Carnegie Mellon University

TATA

Chronic Disease Forecasting

Heart Disease and Stroke Hospitalization Rate Prediction Based on Social Determinants of Health

TCS Team:
Yilun Chen, Tianying Chu, Xiaojun Ma,
Satvika Neti, Ajay Valecha

Abstract

This project is a collaboration between CMU Heinz students in Public Policy Program and Tata Consulting Services(TCS). The focus of project is to provide a predictive model with social determinants of health on heart disease and stroke hospitalization rates at county level. CMU students developed a visual and interactive dashboard to both showcase these predictions of 5 different machine learning models and explore which features might be driving these outcomes. This dashboard enables TCS to identify potential counties with a high hospitalization rate early and intervene with policies related to the driven factors. For codes of this project see the Github Page.

Keywords: hospitalization rate prediction, social determinants of health, machine learning models, interactive dashboard

Heinz College, School of Public Policy & Management
Carnegie Mellon University
May 14, 2021

1 Background

1.1 Problem Statement

Heart disease and other cardiovascular diseases are the number one killer in America, with heart disease causing 655,000 deaths every year and stroke causing 137,000 deaths per year [1]. On top of the human cost, heart disease and stroke cost our healthcare system over $214 billion dollars per year, plus $138 billion in lost productivity on the job [2].

But, if we were able to detect the problem early enough through a risk scoring prediction model, we would be able to intervene with preventative care, specific treatments, or a personalized plan. This would not only increase health outcomes overall, but generate a lower cost to the healthcare industry.

Our client, Tata Consultancy Services (TCS) is a leading company providing software and consulting solutions in the healthcare management. They wish to discover and take actions at the early stage of a chronic disease. Therefore, they hope our team can help them to create a predictive model of cardiovascular diseases.

1.2 Scoping

But how can we create this model? What factors are associated with a higher risk of cardiovascular disease? Early literature review tells us that there are three main categories of risk factors: personal[1][3], such as hypertension, hypercholesterolemia, diabetes, obesity, excessive alcohol use, physical inactivity[4], smoking[5], and depression[6][7]; socioeconomic[8], such as education, income, employment[9], race, poverty, and pollution; and healthcare factors[8] like healthcare approachability, acceptability, availability, accommodation, affordability, and appropriateness.

Based on client feedback, we also decided to focus on predicting not deaths but the hospitalization rate from heart disease and stroke, and scope our population down to just medicare beneficiaries that are 65+. Based on data availability and reproducibility concerns, we decided to focus on prediction at the county level rather than the individual level, and only on counties with populations over 65,000.

Our model would take risk factors such as the social determinants of health and generate a county level prediction for the hospitalization rate of the county for both heart disease and stroke. TCS, our client, would then be able to take these predictions and provide the necessary intervention that would help prevent hospitalizations - and therefore death - from heart disease and stroke.

2 Data Sources

2.1 Outcome

Our outcome variable dataset was relatively easy to find. Based on client feedback, we used the CDC's Atlas dataset[10]. This dataset collects data at the county level for all the counties and for the years 2010-2018. We chose the population to be all races, all genders, and for those over 65. The outcome variable itself is the hospitalization rate (of both heart disease and stroke) on a 3 year average per 1000 Medicare beneficiaries. The Atlas dataset had about a 2% missing rate, but this went down to 0% after subsetting to only counties with a population over 65,000.

2.2 Features

Based on the risk factors in our literature review, we started researching to find data sources for those factors. It was important that each dataset we found had data in the corresponding year of our outcome variable (for example, features in 2010 correspond to outcome in 2011-2013) and was available at all the counties for those years. Some level of missingness was acceptable, but the overall structure needed to be there. The three main datasets we used are below.

2.2.1 American Community Survey

The American Community Survey[11] is an ongoing yearly survey by the Census Bureau with data about demographics, employment, education, and health and much more. This is the main crux of our dataset.

ACS data comes in 2 different formats - 1 year estimates and 5 year estimates. 1 year estimates are only done for counties with a population over 65,000, but are reported yearly. 5 year estimates have all the counties in the US in their dataset, but are only robust estimates of 5 years average. In order to have the same temporal resolution as our outcome variable, we chose to go with 1 year estimates.

Originally, we were going to fill in the missing values with state averages, but we were worried that that would make it difficult for TCS to reproduce this model in the future. After some discussion and client feedback, we decided to limit

the scope of our model to counties that were available in the 1 year estimates. Over 80% of the total US population is compromised of those 834 counties, as well, so while the model cannot generalize to smaller counties, we can still predict for the majority of the US population.

From ACS data, we got variables like employment rate, education attainment, healthcare coverage, and veteran ratio for the 65+ population; race distribution and median income for the total population; and population, gender distribution, and poverty rate for both populations.

Even after subsetting to only the 834 counties, there was still a 9% missing rate in this dataset. To fill those in, we used the state/year average for those counties.

2.2.2 CMS Spending & Prevalence

The CMS Chronic Conditions spending and prevalence dataset[12] provides the level of Medicare spending per county, as well as the prevalence of, a set of about 21 different chronic diseases. This includes heart disease and stroke, but also cancer, hypertension, and drug abuse.

In this dataset, 3 diseases (across both spending and prevalence) had high levels of missingness (36%-82%) - Autism, HIV/AIDS, and Hepatitis. Otherwise the dataset had about a 12% missing rate. Because those three diseases aren't specifically associated with cardiovascular issues (based on lit review), we decided to remove those variables, and then filled in the rest with the recent year's data.

2.2.3 Air Quality Index

AQI, or the Air Quality Index[13], is a dataset that describes how polluted the air currently is. It's been shown that public health risks increase as the AQI increases.

AQI is measured on a scale from 0-500, with 1-100 being good/moderate, 100-200 being unhealthy, 200-300 being very unhealthy, and anything higher being hazardous. This is measured using sensors on the ground in about 1000 counties, that record the presence of 5 major pollutants (ground-level ozone, particle pollution (also known as particulate matter, including PM2.5 and PM10), carbon monoxide, sulfur dioxide, nitrogen dioxide). This dataset is updated twice a year, aggregating daily records into a yearly summary.

From this dataset, we get the overall AQI, the number of bad or unhealthy days, and the amount of particulates in the air for each of those 1000 counties. We also consider the air quality in the previous 8 year to include a long-term effect of pollution.

This dataset had about a 22% missing rate, for counties that didn't have sensors. We once again used the state/year average to impute.

3 Methodology

Our goal was to provide TCS with a visualized, reproducible, and user-friendly solution to predict heart disease and stroke hospitalization rates in the future. Using various machine learning models, we built a dashboard that can impute and preprocess the data, run the models to find the predicted outcomes for hospitalization rates, and then display the predictions of those models and historical trends in both outcomes and features to the user. Figure 1 shows how our dashboard interacts with the whole pipeline based on various user inputs.

First, we had to clean and preprocess the data, along with imputing missing values and combining the datasets together. We then did various methods of EDA to try and find features that were correlated with each other and with the outcome variables and choose the models we will use in the dashboard. We started with a baseline model of linear ridge regression and random forest model to start out.

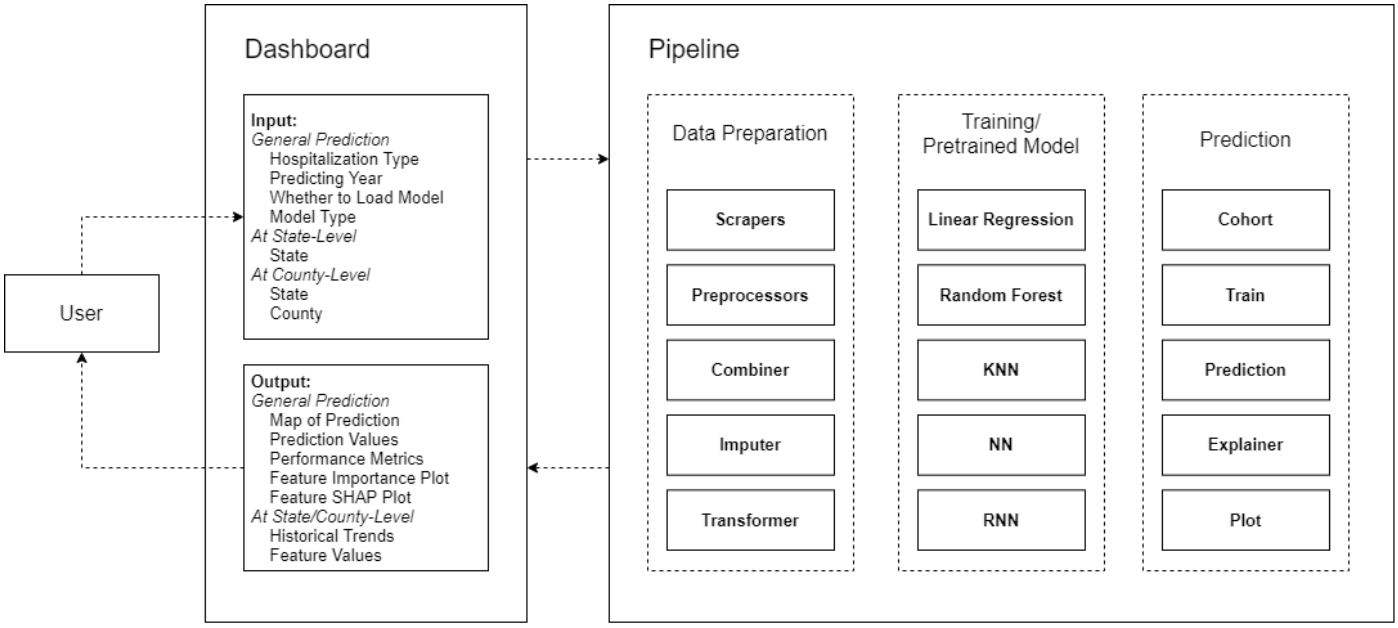
After trying a number of different models, we found that K Nearest Neighbors, Neural Networks, and our Recurrent Neural Network had the highest R^2, so we decided to put them in our dashboard. For each model, we trained on features of 2010 and 2011 to predict on outcome years of 2011-2013 and 2012-2014, respectively.

Using those machine learning models, we created a dashboard where users can choose which model they'd like to predict with, and the output gives you different charts and visualizations to help explain why the trend of prediction might be the way it is. Users can also do a deep dive into a specific state or county and explore the data that way.

This dashboard can be used by TCS to find counties with high hospitalization rates - and especially high outlier hospitalization rates - and explore which features might be driving this change. Or, as we'll see later in the report, could be used to define that it is a feature outside of our dataset that is driving this change.

The rest of this report will go into each of these pieces in detail.

Figure 1: Project Pipeline



4 Exploratory Data Analysis

The exploratory data analysis aims to provide insights for three questions. First, do hospitalization rates for heart disease and stroke vary along the temporal and spatial dimension? A significant temporal variation indicates the necessity to formulate a predictive model and the existence of spatial variation justifies the decision to construct a county-level model. The second question concerns the potential predictive power of the feature variables. Specifically, which feature variation is associated with that of the outcomes? Lastly, to better understand relationships of features across both inter- and inner-datasets, we ask if there exists feature clustering, and thus redundancy by computing the second-order correlation matrix.

4.1 Outcome

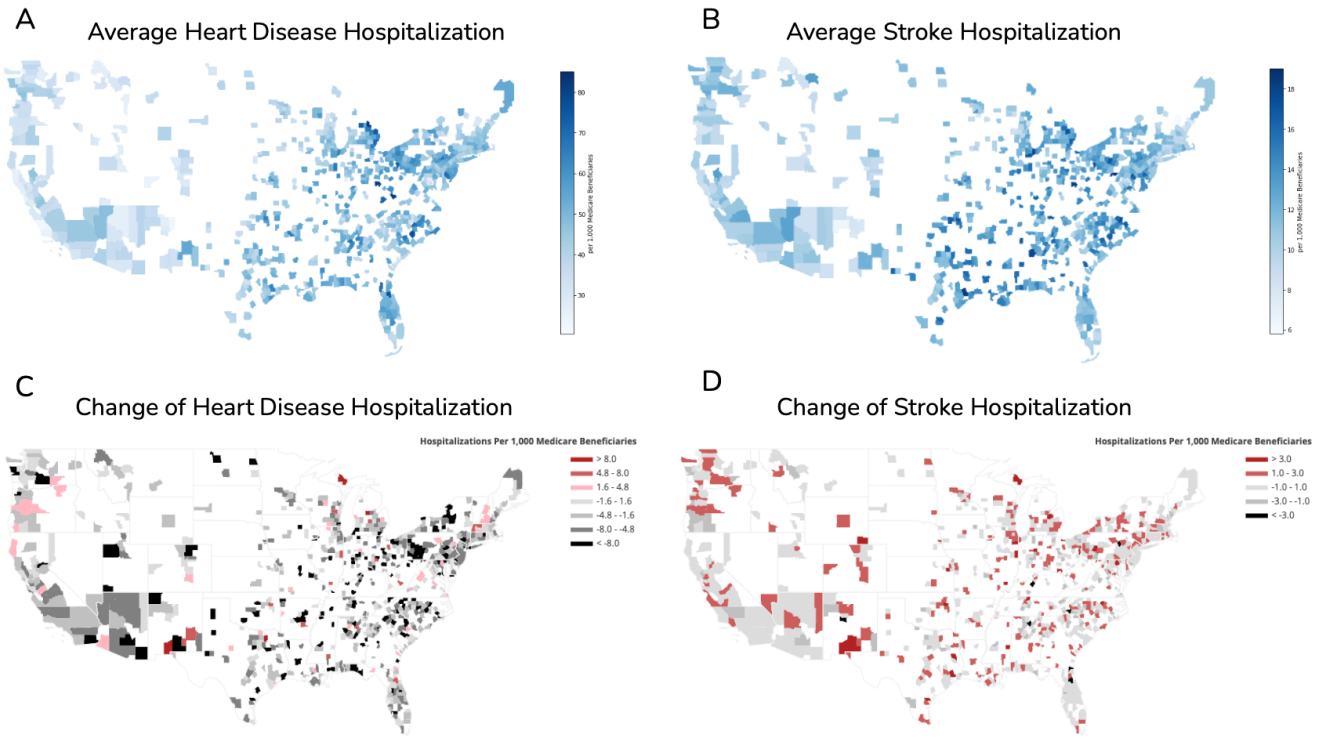
For the temporal variation, we observed that the average hospitalization rates for heart disease decreased by 8.75% from $2011 - 2013$ to $2015 - 2017$ (Figure 2C) while that for stroke increased by 4.00% (Figure 2D). Hence we conclude that a predictive model is needed in order to conduct early intervention. In terms of spatial variation, we found that both heart disease and stroke hospitalization rates are higher in the east than the west. Moreover, average hospitalization rate of heart disease is 297.51% higher than that of stroke. Thus, a county-level model the way we have is justifiable.

4.2 Outcome-Feature

To investigate the relations between features and outcomes, we computed the Pearson correlation for each feature-outcome pairs for both heart disease and stroke. Then, we selected the top 20 features with the highest absolute correlation value for heart disease and stroke hospitalization separately. We found that over 80% of the features in the two lists overlap, indicating that heart disease and stroke have a very similar set of highly correlated features.

We further examine the correlation value in detail and found that the prevalence of a number of chronic diseases are positively correlated with both outcome: chronic obstructive pulmonary disease [.68 - heart disease, .55 - stroke], ischemic heart disease [.67, .47], hypertension [.65, .585], heart failure [.59, .42], stroke [.54, .53], and diabetes [.54, .48]. Also, social determinants such as education $[-.56, -.46]$ and median household income $[-.41, -.34]$ are most negatively correlated with hospitalization.

Figure 2: Spatial and Temporal Variation for Outcome



4.3 Feature-Feature

Figure 3 shows the Pearson correlation matrix for all pairs of features grouped by datasets. We found that features in the chronic disease spending dataset are highly correlated with each other. Similarly, we observed correlation clusters within the air quality index dataset, socio-economic factors, and chronic disease prevalence. Other than the chronic disease spending and prevalence, we didn't observe high correlation among variables belonging to different datasets. Therefore, we conclude that in general, different datasets capture different dimensions of information while there exists feature redundancy within each datasets.

5 Model

5.1 Model Formulation

For each county with 65,000+ population, every year, we will work to predict, for the next 3 years, the average hospitalization rate (per 1,000 Medicare beneficiaries) for 65+ patients. We are using prediction years 2010 and 2011 (with outcome years 2011-2013 and 2012-2014, respectively) as our training set. We randomly split 50% of the data in 2014 (with outcome years 2015-2017) as our validation and test set so that there is no overlap between our outcome years. Our final prediction will be for prediction year 2018, to predict the outcomes for 2019-2021.

5.2 Model Training

First, we standardized continuous features using Equation (1), where x is feature value, μ and s are the the mean and standard deviation, respectively, of the feature within the training sample. For neural network and the recurrent neural network, we scaled the outcome variables such that they are from 0 to 1.

$$z = \frac{(x - \mu)}{s} \tag{1}$$

Figure 3: Feature-Feature Pearson Correlation

5.2.1 Metrics Selection

Our predicted outcome is the hospitalization rate, which is a continuous value. To measure the model prediction of a continuous outcome, we have chosen to use unadjusted R^2 as our primary metrics and other metrics such as mean squared error, mean absolute error as supporting metrics. The unadjusted R^2 measures how much of the outcome variance is captured by the model, it is a value bounded by 0 and 1 (in special cases where the prediction is worse than predict with mean, it can be negative). Using the R^2 has the benefit of:

1. As a scaled metrics, it is easy to compare performance of different models, the closer the R^2 is to 1, the better the model performs.
2. The meaning of R^2 is easy to understand, as mentioned above, it measures how much of the variance is captured.
3. R^2 is a commonly used metrics for continuous value prediction.

As a result, we chose to measure our model performance primarily with unadjusted R^2, and using other metrics as supporting metrics.

5.2.2 Validation and Test Method

Due to the limited data available, the cohorts (feature + outcome pair) we have are the cohort 2010 (2010 feature + hospitalization average of 2011-2013), 2011, 2012, 2013 and 2014 (2014 feature + hospitalization average of 2015-2017). Since the outcome is a three year average, we want to prevent the overlap of outcome year range between our training set and validation set. As a result, we are left with very limited choices, and must train with cohort 2010+2011 and validate with cohort 2014. In this case, the outcome of training set ranges from 2011 to 2014 and outcome of validation set ranges from 2015-2017, and there is no overlap between.

Ideally, we want to similarly have another test set that is 3 years apart from the validation set for testing models, but due to limited data, we don't have that. So, the final idea we came up with is to random split the 2014 cohort into two subparts, one for validation and use that one for hyper-parameter tuning, another one for test and use that to test the model performance, get the performance metrics. In this way, there are no time overlap between training set and test set, and no overlap between training set and validation set. Since the validation set and test set are not the same, we eliminate the risk of overfitting our model to validation set during the hyper parameter tuning. In all, this is the best way we can think of to get validation and test set with the extremely limited data we have.

5.2.3 Ridge Regression

We use the ridge regression as our baseline model here. Ridge regression is a variant of the simple linear regression. The difference is that ridge regression includes a *L2* norm regularization term to regularize the model and prevent it from ovrefitting. The simple ridge regression model is a baseline model since it shows that what is the performance simple models could do, so we will use the ridge regression performance as a benchmark for our more complex models.

After traning and testing, the ridge regression is able to achieve a R square of 0.44 for heart disease hospitalization prediction, a R square of -0.08 for stroke prediction. We can see that the simple regression does not perform well for our task, especially for the stroke prediction. A negative R^2 is not normal in machine learning models because that means the model prediction is worse than just predict every data point with a constant number. However, in our case, besides showing that more complex models are needed, it also shows us that stroke prediction is more difficult and the stroke hospitalization is probably less predictable. With that, we are using the ridge regression to get some quick insights and form a baseline that our more advanced models want to beat. The more advanced models we try later do significantly outperform the ridge regression.

5.2.4 Random Forest

Random forest is an ensemble method that can be used for both classification and regression. The method uses random subset of features and data points to train decision trees, and then aggregate the result of multiple decision trees to get the final prediction. In this way, random forest correct for decision trees' habit of overfitting to its training set, and generally shows better performance than a single decision tree. Besides good model performance, random forest as a tree-based model also has the benefit of being easily interpretable, meaning the model itself is able to give back feature importance and model structure to help understand the model prediction process. The benefit of good performance and interpretability makes random forest and its variants one of the most popular machine learning methods.

In our case, we are using random forest regressor to predict the hospitalization rate for heart disease and stroke separately. The model is trained using 2010 and 2011 cohort, validated and tested with the 2014 cohort. After hyper-parameter tuning, we decide to use random forest regressor with 20 ensemble trees, no max-depth and MSE criterion. The random forest regressor proves to give pretty good performance: the R^2 for heart disease hospitalization prediction on test set is 0.68, for stroke hospitalization prediction is 0.39.

5.2.5 K Nearest Neighbors

K-nearest neighbor (KNN) is a non-parametric algorithm and can be used for both classification and regression problems. The KNN algorithm uses feature similarity to predict the values of any new data points. This means that the new point is assigned a value based on how closely it resembles the points in the training set. KNN algorithms can be easy to implement as they don't need to make assumptions about the dataset and can work with non-linear data; however, as the dataset grows, efficiency and speed of the algorithm declines very fast.

In order to predict both hospitalization rates for heart disease and stroke, we trained the KNN models on data from 2010 and 2011, so that we can avoid the overlap in the outcome variable. For hyperparameter tuning, especially the number of k, we split the data from 2014 into two and used one set to tune the model and the other to test our models. Based on the results of hyperparameter tuning, the best results were achieved with number of k being equal to 3 and similarity metric being the Manhattan distance.

Our evaluation metric for model accuracy is R^2. The resulting R^2 score on test data for KNN models are 0.67 and 0.45 for heart disease and stroke respectively. These results are quite similar to what we got for Random Forest.

5.2.6 Neural Network

Neural networks allow for constructing highly flexible, complex model structures as well as adopting parametric, continuous activation functions to learn higher-order feature characteristics. We first fit an unidirectional, feed-forward neural network with 1 hidden layer and 20 hidden units. As shown in Figure 4A, the input layer contains standardized features in year 2010 and 2011 and maps them to the internal hidden representation with the nonlinear activation function - hyperbolic tangent ($tanh$) whose codomain is from -1 to 1. Then, the hidden representation is further mapped to the output layer with one unit representing the scaled hospitalization rate. The *sigmoid* function whose codomain ranges from 0 to 1 was used as the output function.

The loss function ($\mathcal{L}_{NN}$) is defined in Equation (2), where N is the number of training data points, y_i and $\hat{y}_i$ are the actual and predicted hospitalization rate for the $i-th$ data point respectively, γ is the regularization parameter, K is the number of model parameters (weights), and ω_j is the $j-th$ weight. The first term of $\mathcal{L}$ is the standard MSE loss and the

second term is the $L-2$ regularization whose function is to combat model overfitting. We used the Stochastic Gradient Decent (SGD) algorithm for optimization and stopped the training process when SGD converged.

$$\mathcal{L}_{NN} = \frac{1}{N}\sum_{i=0}^{N}(y_i - \hat{y}_i) + \gamma\sum_{j=0}^{K}\omega_j^2 \tag{2}$$

We optimized model hyper-parameters systematically using the validation dataset. In particular, we tuned the learning rate, activation function, momentum, and the regularization weight, and selected the model with the lowest validation error. Furthermore, we increased model complexity by adding up to 3 hidden layers and up to 50 hidden units, and repeated the same hyper-parameter-tuning process. By making the structure more complex, the best validation error decreased by at most 3.33% for heart disease and 1.65% for stroke and the un-adjusted R^2 remains the same for heart disease and stroke in the best case. Thus, concerning the drawbacks of a complex model, in particular the higher chance of overfitting as well as the higher computation cost, we decided to go with the model with simple structure - the original setting of 1 hidden layer and 20 hidden units.

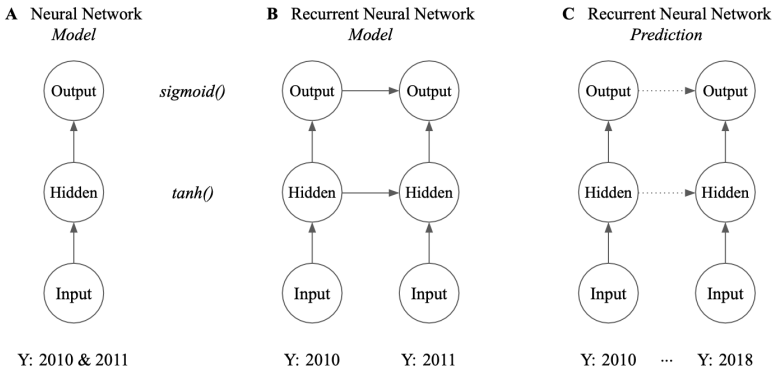
5.2.7 Recurrent Neural Network

Recurrent neural network can potentially model the temporal dependency that's existing in the data. Due to the limited time range of the outcome variables, however, we were only able to model the temporal dependency using 2-year training data. Figure 4B demonstrates the architecture of the recurrent network. Similar to the neural network, we used 1 hidden layer with 20 hidden units. In the training process, the model will first carry out the forward path using features from 2010. The hidden representation of year 2011 depends on both the feature of 2011 as well as the hidden representation of 2010. Similarly, the output of 2011 depends on the hidden representation of 2011 as well as the output of 2010. When making prediction for any given year Y, we constructed a 3-dimensional tensor incorporating features from 2010 to Y, then computed $Y-2009$ outputs for each year, and finally evaluated the model performance using only the prediction in year Y. The activation and output functions for the recurrent network are the same as those of the neural network described in the previous section.

Using the similar notation, the loss function ($\mathcal{L}_{RNN}$) is defined in Equation (3), where T is the total number of years in the training data. The training and hyper-parameter tuning processes were the same as those described in the previous section for training neural network.

$$\mathcal{L}_{RNN} = \frac{1}{T\times N}\sum_{t=0}^{T}\sum_{i=0}^{N}(y_i^t - \hat{y}_i^t) + \gamma\sum_{j=0}^{K}\omega_j^2 \tag{3}$$

Figure 4: Neural Network and Recurrent Neural Network



5.3 Model Summary

Table 1 summarizes the un-adjusted test R^2 for all of models we trained for both heart disease and stroke hospitalization. In terms of heart disease hospitalization, machine learning models slightly outperform the linear baseline model - ridge regression. For stroke hospitalization, however, machine learning models realized a dramatic improvement in terms of accuracy, indicating that non-linearity is indeed critical to capture the outcome variation. Furthermore, the recurrent

neural network significantly outperforms other machine learning models in this case, which indicates that modeling the temporal variation of the data may be essential to model stroke hospitalization.

Model	Heart Disease R^2 (test, un-adjusted)	Stroke R^2 (test, un-adjusted)
Ridge Regression	0.44	-0.08
Random Forest	0.68	0.39
K-Nearest Neighbors	0.67	0.45
Neural Network	0.69	0.22
Recurrent Neural Network	0.71	0.52

Table 1: Summary of R^2 for 5 models

6 Dashboard User Cases

To help users better understand our model's result, we developed a dashboard that can run the models based on users' requirements and display detailed graphs. All the graphs can be reproduced yearly with the latest data updates as needed.

Users can select: whether they are interested in heart disease or stroke hospitalization; which year they would like to predict; whether to use a pre-trained model or not; and which type of model they would like to use. These four inputs will then be sent to our pipeline to make the predictions.

Within the pipeline, we have 4 scrapers to collect data from the four data sources and 4 pre-processors to format the raw data. Then the raw data from all four sources are joined together by a combiner, and sent to the imputer to input the missing data and transformer to transform the data into a readable format for our models.

Next, with all the data prepared, we implement the model that the user inputted. We created 5 models in total. For each model, the pipeline will first get the corresponding cohorts, and then the model will be trained on the existing data and predicted for the year the user input. If the user chooses to use a pre-trained model, then training process will be skipped and the predicted values from the pre-trained model will be given.

Finally, all figures and tables are plotted using the prediction data. The first tab of the dashboard provides the table of predicted values, a map visualization of the predicted values, model performance metrics, a feature importance plot, and a feature contribution plot using SHAP values(for the explaination of SHAP value, see Appendix A.2.2).

To get state and county specific analysis, users can input either a state in the second tab or a state and county in the third tab. The dashboard will provide historical trends of the hospitalization rate along with the predictions. The users can further explore the causes behind the predicted values by investigating feature trends.

This section will include 3 informative, intuitive use cases of our dashboard. For further explanations about how the dashboard is structured and how to read each graphs in the dashboard, please see the user manual in the appendix.

6.1 Top Features

We visualize the top 10 most important features for each model in our dashboard. Overall, overlap of top 20 features for heart disease hospitalization rate from RF, NN, RNN are shown in Table 2, and we can see that features like income and educational attainment contribute negatively to the heart disease outcomes, and prevalence of other chronic diseases like hypertension, COPD, and diabetes have positive contribution. We also see that for the RNN, the past year's predictions seems to be a positive factor as well.

6.2 Early Intervention

Our dashboard has many use cases, and pinpointing places for early intervention is one of them. For example, we can see in Figure 5 that *Creek, Oklahoma* is an example of a county where the predicted values follow pretty closely with the actual values, but then after, we see a dramatic increase in model prediction after year 2015-2017.

What features are driving this change? And what could policy makers do about it? We can see below in Figure 6 that our socio-economic factors don't explain this increase, but the prevalence of ischemic heart disease and heart failure followed a similar trend to hospitalization, and there was a dramatic increase in hypertension and diabetes starting around 2011. Through this analysis, we see that these early risk factors might be the early signs of both heart disease prevalence and hospitalization. Policy makers could then use this information to intervene - perhaps by creating new laws or working to educate people on the dangers of these early risk diseases and how to prevent them.

Features	Negative or Positive Correlation
Social Determinant	
Household Median Income	Negative
Education Attainment	Negative
Percentage of Male	Negative
Other Chronic Disease	
Ischemic Heart Disease Prevalence	Positive
COPD Prevalence	Positive
Heart Failure Prevalence	Positive
Hypertension Prevalence	Positive
Diabetes Prevalence	Positive
Hyperlipidemia Prevalence	Positive
Temporal Dependency (RNN Only)	
Previous Output	Positive

Table 2: Common Set of TOP 20 Important Features for Heart Disease Hospitalization

6.3 Outlier Exploration

We can also use our dashboard to investigate outlier counties. Scioto County in Ohio is one of those outliers - in both a high heart disease hospitalization rate and a high deviation between the predicted and actual values.

The actual values of heart disease hospitalization as seen in Figure 7 first decreased from 2011-2013, and then dramatically increased afterward. None of our models were able to accurately predict this increase, which leads to Scioto being an outlier in model performance. This points to there being potentially features that are driving this change that are not in the datasets we are working with.

The same important features we looked at earlier could help explain why Scioto is such an outlier - as we see in Figure 8 it deviates far from the national average in all of these features, and none of these features have the same abrupt increase that the hospitalization rate did. Therefore, our models are able to predict a baseline outcome for Scioto, but can't predict an abrupt change being driven by features not in our dataset. In this way, our dashboard can help to isolate outliers, narrow down what's causing these changes, and encourage further investigation into what happened to drive this change.

6.4 Recommendations

Based on our analysis and exploration of the data, we can see that income, education attainment, and medical risk factors like hypertension and COPD are high risk factors for heart disease and stroke hospitalization rates. Early intervention could be based on looking at the model prediction and investigating feature trends to explain the change, and outlier exploration could help identify areas of more research, or - if it's an outlier with a lot fewer hospitalizations, can help us learn from good policy interventions.

7 Discussion

7.1 Limitations

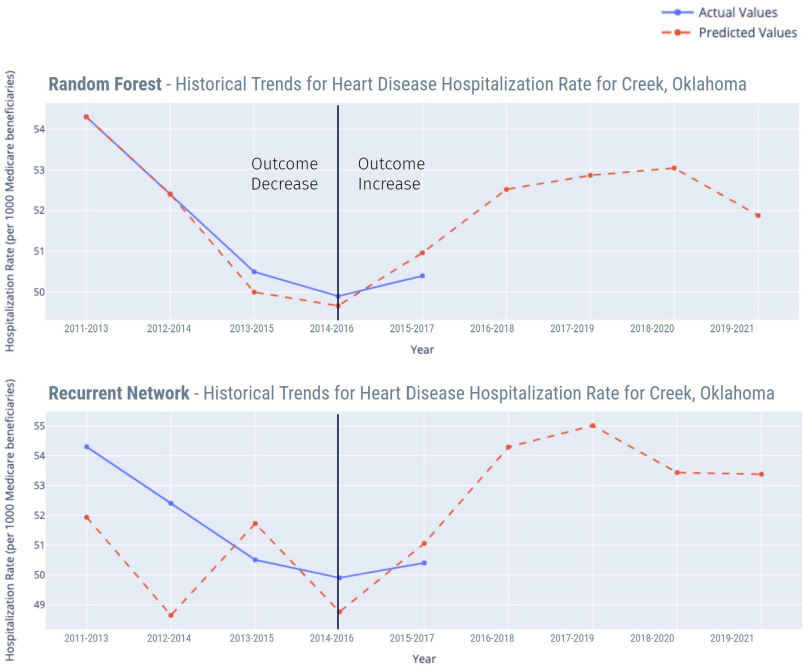
There are many limitation to this solution, and they fall under three main buckets: access to data, modeling, and implications of the results.

7.1.1 Incomplete Access to Data

1. There is no data available for chronic diseases on individual level due to privacy and legal concerns involved in healthcare records. Therefore, it was not possible to build the predictive risk scoring model to offer personalised care plan on an individual level.

2. Due to the specific data sources used in this project, the data was not available for all 3,142 counties in the U.S. The data was available for only around 800 counties, therefore the model predictions are limited to those 800 counties. One thing to notice is that those 800 counties have over 65,000 population each, covering over 80 percentage of total population of the U.S.

Figure 5: Creek, Oklahoma Hospitalization Trends



3. Although, we had access to data on chronic disease prevalence and spending, we did not have healthcare accessibility data available to use in our models.

4. There is uncertainty over the release of data in the future from different data sources used in this project. For example, CMS has data on prevalence and spending until 2018. We do not know when will they release the data for 2019 and onward, which makes our models contingent on the availability of data from these data sources

7.1.2 Modeling

1. Hyperparameter tuning is an essential component of building machine learning models. We need a validation dataset, which is different from train and test dataset, in order to tune the ML hyperparameters so that our models do not overfit. However, we did not have a validation dataset. We got around that problem by splitting the 2014 cohort data (feature year) into validation and test data to tune our hyperparameters for the models, but a true validation set would help our model performance further.

2. Since the outcome variable, hospitalization rates, is a number in a given range of years, there was inability to train data from all available years due to temporal overlap in the outcome variable. For instance, we could not use data from 2013, because its outcome variable, hospitalisation rates 2014-2016 overlapped with hospitalisation rate 2013-2015 from 2012 cohort data (feature year).

3. Due to potentially important features missing in our data, prediction results for stroke hospitalisations remained consistently substandard for all of our machine learning algorithms.

7.1.3 Implications

1. We have to be careful with the interpretation of our results, for they are based on correlation and do not imply causation. The results shown of important features are based purely on correlation and predictive power, but important are not casual. Further research must be done on causality.

7.2 Going Forward

Tata Consultancy Services can start an early intervention program based on the prediction results of our model for specific counties in the U.S. whose hospitalizations rates differ significantly from national or state averages. Furthermore,

Figure 6: Creek, Oklahoma Feature Trends

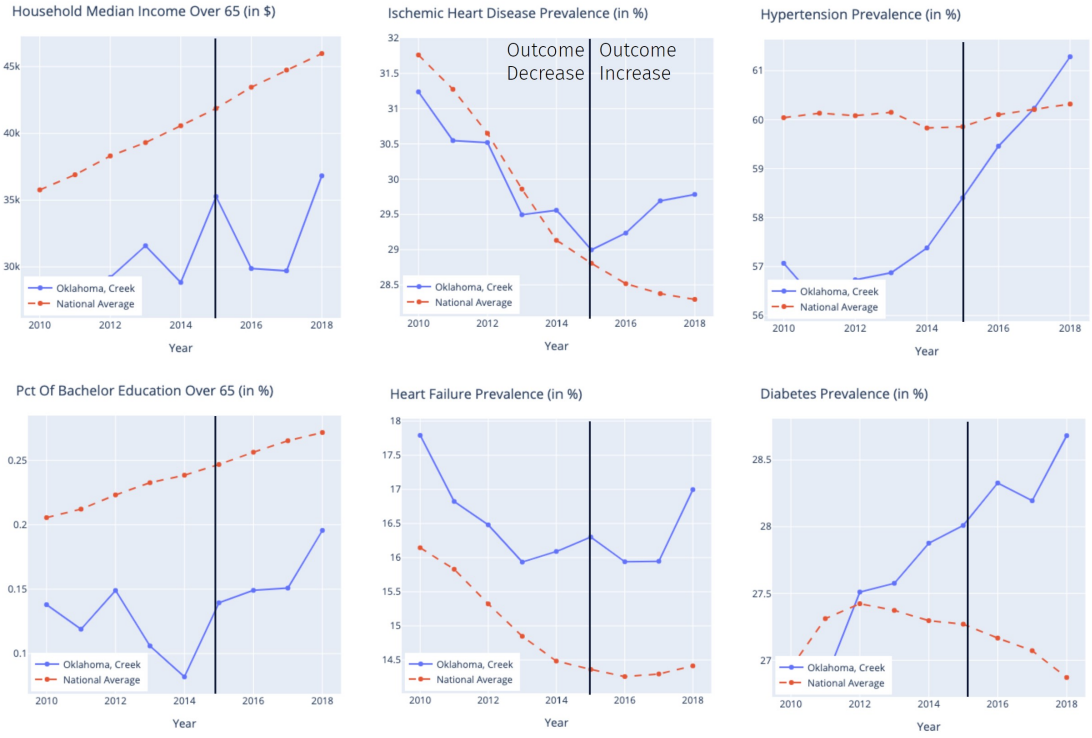
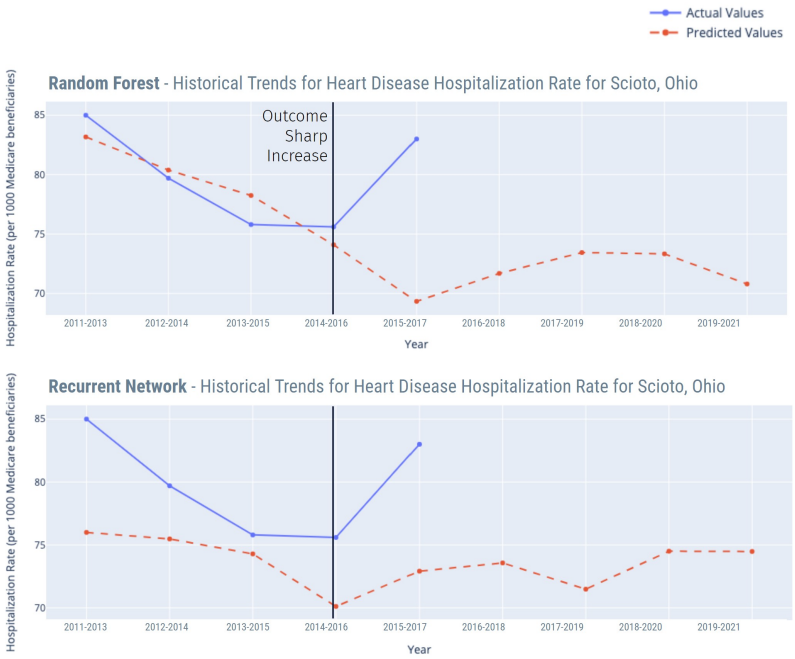


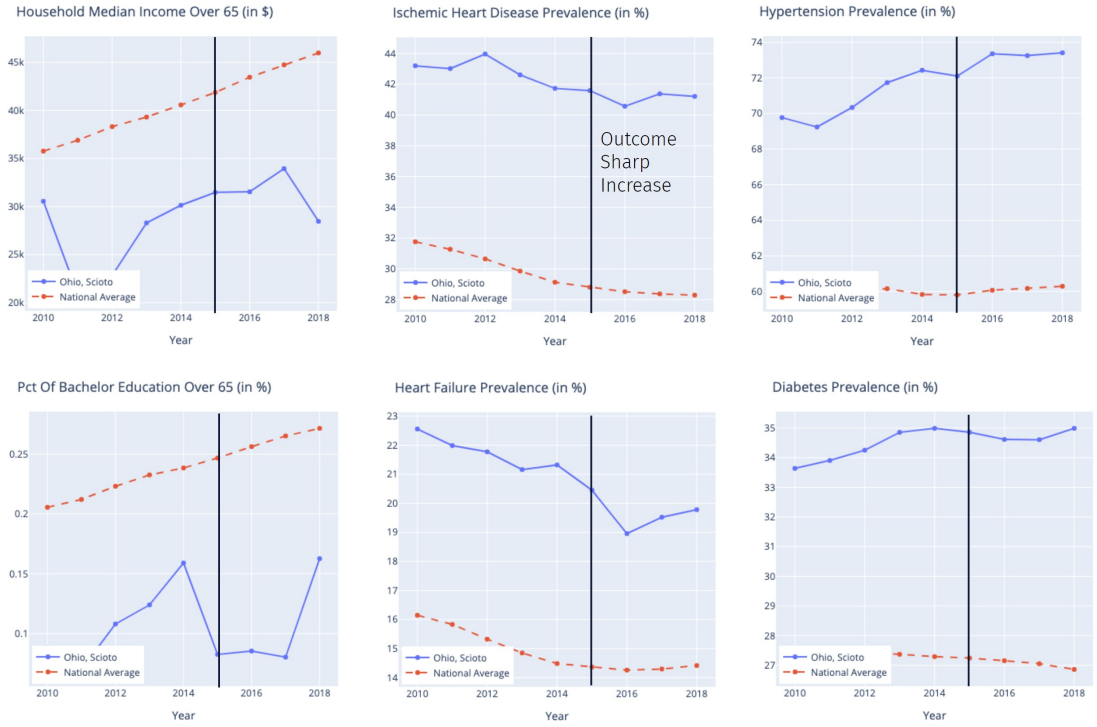
Figure 7: Scioto, Ohio Hospitalization Trends



they can dive deep and analyze feature causality for the top features recognised by our models. That would help in explaining and understanding the unexpected shocks observed.

The future teams working on the continuation of this project could find data on more relevant risk factors to further improve the model results, especially stroke hospitalization rates. Moreover, they can figure out how to streamline the pipeline to access the data from the data sources without any looming uncertainty around data availability. Finally, once

Figure 8: Scioto, Ohio Feature Trends



the project is completed and finalised, they can publish the results and extend the application and usability of the product to general healthcare community.

8 Conclusion

The problem we were trying to solve was predicting hospitalization rates at the county level for heart disease and stroke, and finding ways to see what was driving that change for policymakers and TCS to be able to do early preventative intervention.

We used 4 different data sources, including our outcome dataset, and 5 different machine learning models. We then created a visual, interactive dashboard so that our client would be able to not only predict these values for the future in an intuitive, easy to use way, but also visualize different features to see how our features impact the hospitalization rates.

We hope that further exploration can work on casual analysis rather than just correlation or predictive power, and work to include more features.

References

[1] Heart disease and stroke statistics—2020 update: a report from the american heart associationexternal icon, Circulation 141 (9) (2020) e139–e596.

[2] Prevalence of uncontrolled risk factors for cardiovascular disease: United states 1999–2010, NCHS data brief 103.

[3] Racial and ethnic differences in cardiovascular disease risk factors: A systematic review, Ethnicity Disease 17 (2007) 144–152.

[4] Physical fitness and activity as separate heart disease risk factors: a meta-analysis, Med Sci Sports Exerc 33 (5) (2001) 754–761.

[5] Mortality risk reduction associated with smoking cessation in patients with coronary heart disease, JAMA 290 (1) (2003) 86–97.

[6] Generalized anxiety disorder prevalence and comorbidity with depression in coronary heart disease: A meta-analysis, Journal of Health Psychology 18 (12).

[7] New insights into the comorbidity of coronary heart disease and depression, Current Problems in Cardiology 46 (3) (2021) 100413.

[8] Social determinants of risk and outcomes for cardiovascular disease, Circulation 132 (3) (2015) 873–898.

[9] The cumulative effect of unemployment on risks for acute myocardial infarction, Arch Intern Med 172 (22) (2012) 1731–1737.

[10] CDC (Centers for Disease Control) Atlas Data[link].
URL https://nccd.cdc.gov/DHDSPAtlas/Reports.aspx

[11] ACS (American Community Survey) Data[link].
URL https://www.census.gov/data/developers/data-sets/acs-1year.html

[12] CMS (Centers of Medicare and Medicaid Services) Spending and Prevalence Data[link].
URL https://www.cms.gov/Research-Statistics-Data-and-Systems/Statistics-Trends-and-Reports/Chronic-Conditions/CC_Main

[13] AQI (Air Quality Index) Data[link].
URL https://aqs.epa.gov/aqsweb/airdata/download_files.html#Annual

Appendices

A User Manual

The purpose of this User Manual is to:

- help users of Dashboard to learn how to install the prerequisites in their computer and run the dashboard by themselves.
- explain all the graphs and tables detailedly and discuss how we get the numbers in the graphs.

A.1 Prerequisites and Installation

1. Operating System: Windows 10, 64-bit operating system
2. Programming Language: Python 3.8.3 (if you don't have python installed on your computer. Please download Python from this website)
3. Prerequisite Library:
   - Basic library: pandas, numpy, os, re, zipfile, joblib
   - Web scraping library: selenium, censusdata
   - Machine learning library: sklearn(sci-kit learn), torch, shap
   - Dashboard library: dash, plotly, matplotlib

   If any of the libraries is not installed on your computer, try to install them in the way suggested by this website.
4. Chromedriver: To run our dashboard, you need to have a Chrome browser on your computer and download the corresponding version of Chromedriver from this website.

A.2 Dashboard Content

At a high level, our dashboard includes a **User Input Interface** and 3 tabs: **General Prediction**, **State-Level Analysis**, and **County-Level Analysis**. We will use the prediction of *heart disease hospitalization rate in 2019-2021 by the pre-trained RNN model* as the example in the following paragraphs.

A.2.1 User Input Interface

Figure 9 displays the user input interface in our dashboard. It begins with a brief introduction about our dashboard. Then users can select **what hospitalization rate they are interested in**(*heart disease* or *stroke*), and **which year they would like to predict**. Our dashboard would check whether all the data sources are available to make the prediction of target year. Prevalence and spending data from CMS are not published at a regular basis. Thus, our dashboard allows users to upload the latest CMS data by themselves.

If all the data sources are available, users can choose a **type of model** to make prediction. Users can either **use a pre-trained model or train the model in the real time**. The pre-trained models are trained with all the available data at current time. But users can also choose to train a new model with the updated data in the future. There are 4 models connected to our dashboard: *Linear Regression*, *Random Forest*, *Neural Network* and *Recurrent Neural Network*. *K Nearest Neighbors* is not listed here because it's very time-consuming to be explained by SHAP kernel explainer.

A.2.2 General Prediction

The "General Prediction" tab includes 5 components: US Map of Prediction, Table of Prediction, Model Metrics, Feature Importance Plot, and Feature Contribution Plot with SHAP values.

Figure 10 is a **national map of our predicted hospitalization rate**. The lighter the color is, the higher hospitalization rate a county has. The prediction is the hospitalization rate per 1000 Medicare beneficiaries in the target year. We also displays the predicted hospitalization rate in the table in our dashboard (See Figure 11). It helps users to rank counties by their predicted hospitalization rate or search for the prediction for a specific state/county. For example, *Scioto, Ohio* is the county predicted to have the highest hospitalization rate in 2019-2021 by RNN model.

Figure 9: User Input Interface

CMU X TCS-Chronic Disease Forecasting

- Heart Disease & Stroke Hospitalization Rate Prediction

Introduction

> The dashboard predicts hospitalization of heart disease and stroke. The outcome variable comes from CDC Heart Atlas dataset. It represents hospitalization rate per 1,000 Medicare beneficiaries (3-year average). Features are social determinants of health, including American Community Survey (ACS), CMS - Chronic Conditions Spending and Prevalence, and EPA Air Quality Index.
>
> Before running the model, please first select `Hospitalization Type` and `Prediction Year` to check and update data to the latest year. If the prediction year is after "2019-2021", please upload the latest CMS data.
>
> Then you can choose `Whether to Use Pre-trained Model` and `Model Type`, and run the model. You can see the prediction and model performance on the "Prediction" tab.
>
> If you are interested in the state-level analysis or county-level analysis, please select `State` and `County` on the "State-Level Analysis" tab and "County-Level Analysis" tab. You can also choose the features you are interested in within the state/county on these two tabs.

Upload latest CMS data(if necessary):

Drag and Drop or Select Files

No file uploaded.

Hospitalization Type: Heart Disease

Prediction Year: 2019-2021

Check Feasibility: CHECK YEARS

Successfully update data to the latest year!

Use Pre-trained Model: Yes

Model Type: Recurrent Neural Networks

Run model: SUBMIT

Heart disease hospitalization in 2019-2021 with RNN model is successfully predicted!

Figure 10: US Map of Prediction

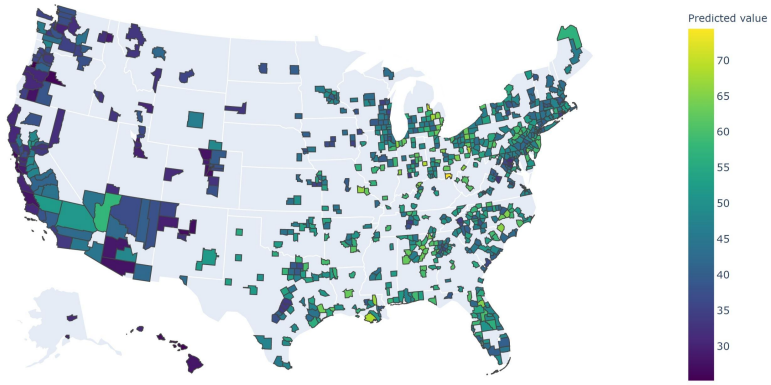


Figure 11: Table of Prediction

Table of Prediction

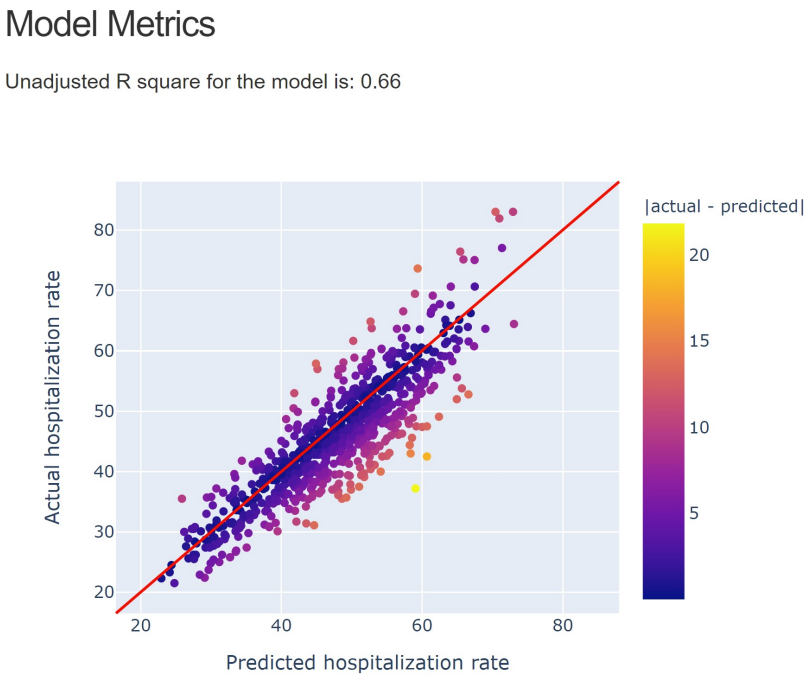
DOWNLOAD CSV

fips	state	county	predicted hospitalization r
filter data...			
39145	Ohio	Scioto	74.49
26017	Michigan	Bay	70.22
18177	Indiana	Wayne	68.98
39081	Ohio	Jefferson	68.93
22109	Louisiana	Terrebonne Parish	68.55
11001	District of Columbia	District of Columbia	68.53
26049	Michigan	Genesee	67.64
22105	Louisiana	Tangipahoa Parish	67.01
29071	Missouri	Franklin	66.61
18167	Indiana	Vigo	66.17
18089	Indiana	Lake	65.93
37155	North Carolina	Robeson	65.81
54107	West Virginia	Wood	65.72
18019	Indiana	Clark	65.57
26087	Michigan	Lapeer	65.43

« ‹ 1 / 51 › »

To evaluate how well a model perform, we provide a **scatter plot of "Actual Value vs. Prediction"** along with an un-adjusted R^2 (see Figure 12). Prediction of counties on the diagonal (in a darker color) are more accurate than those away from the diagonal (in a lighter color). If the points are concentrated on both sides of the diagonal, then the performance of the model is pretty well.The R^2 is based on the validation set (the last year we have actual hospitalization rate values) and provides users a quantitative way to measure model performance. For example, our RNN model has an un-adjusted R^2 of 0.7 on the validation set, which means it can explain 70% of the variance in the hospitalization rate.

Figure 12: Model Metrics



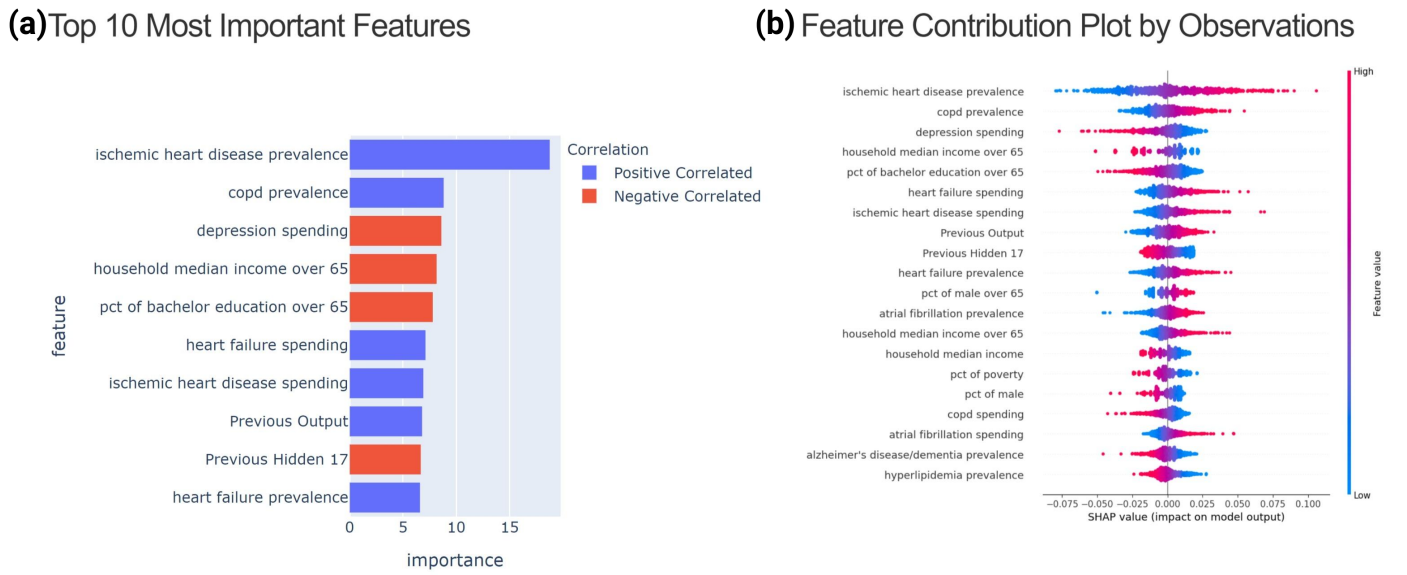
It's also important to know what features result in the prediction of our models, so we create two feature importance plots in our dashboard (see Figure 13). These two plots both rely on the SHAP value of observations. The SHAP values

is used to quantitatively measure the influence of each feature to the predicted outcome. It is a popular method used to understand feature importance for complex models. We can simply regard the SHAP value of a feature as how much this feature has contributed to the outcome. Here, We have a SHAP value for each feature in each observation.

The left panel (Figure 13.a) is a bar plot that shows the **top 10 most important features** in an aggregated level. It means that across all the predicted counties, the listed features are the ones with greatest influence on the output. To be specific, each feature's 'Importance' here is calculated by summing the absolute SHAP values of that feature for all the counties. Bigger 'Importance' value means the feature is more important. To further understand the relationship between the feature and the outcome, we want to have an idea in which direction the feature is driving the outcome to. So, we color the bars by the sign of correlation between feature values and SHAP values. So, the bar colors indicates whether the feature is positively or negatively correlated with the outcome.

The right panel (Figure 13.b) is a violin plot that looks at **feature contribution at the single county level**. Each point represents an observation of county-year combination. The x-axis is the SHAP value of features. We can see how the SHAP values are distributed for each feature listed. The color indicates the value of feature itself. Red points are observations with larger feature values and blue points are observations with smaller feature values. For features like *ischemic heart disease prevalence*, the SHAP values can vary greatly by observations, and a higher ischemic heart disease prevalence will lead to a higher SHAP value.

Figure 13: Feature Importance Plots



A.2.3 State-Level/County-Level Analysis

The "State-Level Analysis" and "County-Level Analysis" follows a similar logic. So we will explain this two tabs together. Both tabs include a plot of historical trend for the prediction outcome and 3 similar plots of feature's trend.

Figure 14 and 15 are the **historical trend for the outcome, hospitalization rate**. The x-axis is the year of prediction outcome and y-axis is hospitalization rate per 1000 Medicare beneficiaries. The blue line is the actual hospitalization rate we have. Currently, we only have actual hospitalization rate to 2015-2017 three year average. The red dash line is the predictions we make. Given features available at 2018, we are able to make predictions of 2019-2021. And the green dash line is the national average of our predictions. For "County-Level Analysis", the actual and predicted hospitalization rate is the value of users' interested county, and the national average is the average of all the counties in each year. For "State-Level Analysis", the actual and predicted hospitalization rate is the state average of counties within the interested state, and the national average is the average of state average in each year. The purpose of these two plots is to help users find states and counties with an increasing hospitalization rate in the future year and start early interventions for those states and counties.

Figure 14: County Historical Prediction Trend

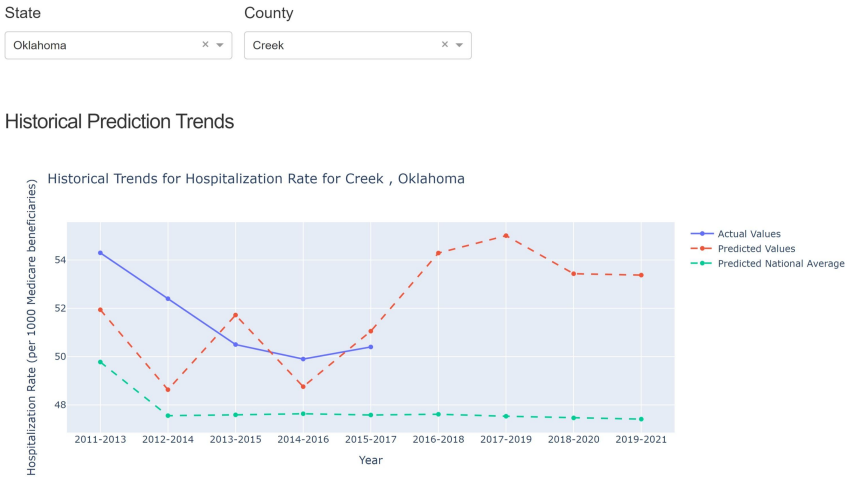
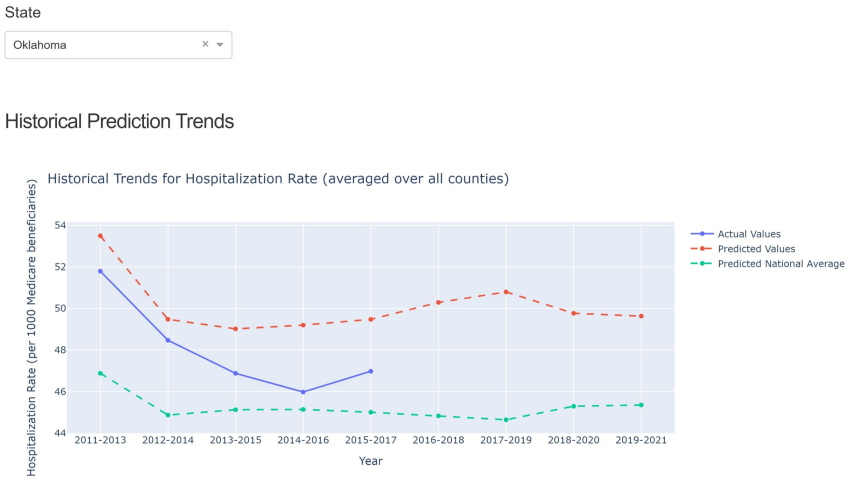


Figure 15: State Historical Prediction Trend



Our dashboard also allows users to further explore **the trend of features** and how the feature values impact our prediction of a certain state or county. So if a state or county has an increasing trend of hospitalization rate, users can know what risk factors lead to that increase. Figure 16 and 17 plot how the feature values change over time. The blue line is the actual feature values of the state/county and the red line is the national average for the feature. Users can learn what level the state/county is in the country. The aggregating methods of national average is more complicated for features. For available features and aggregating methods, please see Table 3.

Features	State National Average	County National Average
ACS-Median Income	Simple average of state average	Simple average of all counties
ACS-Population	Simple average of state average	Simple average of all counties
ACS-Others	Population-weighted average of all counties	Population-weighted average of all counties
CMS-Spending	Population-weighted average of all counties	Population-weighted average of all counties
CMS-Prevalence	Population-weighted average of all counties	Population-weighted average of all counties
AQI	Simple average of state average	Simple average of all counties

Table 3: Aggregating Methods for National Average

Figure 16: County Features Trend

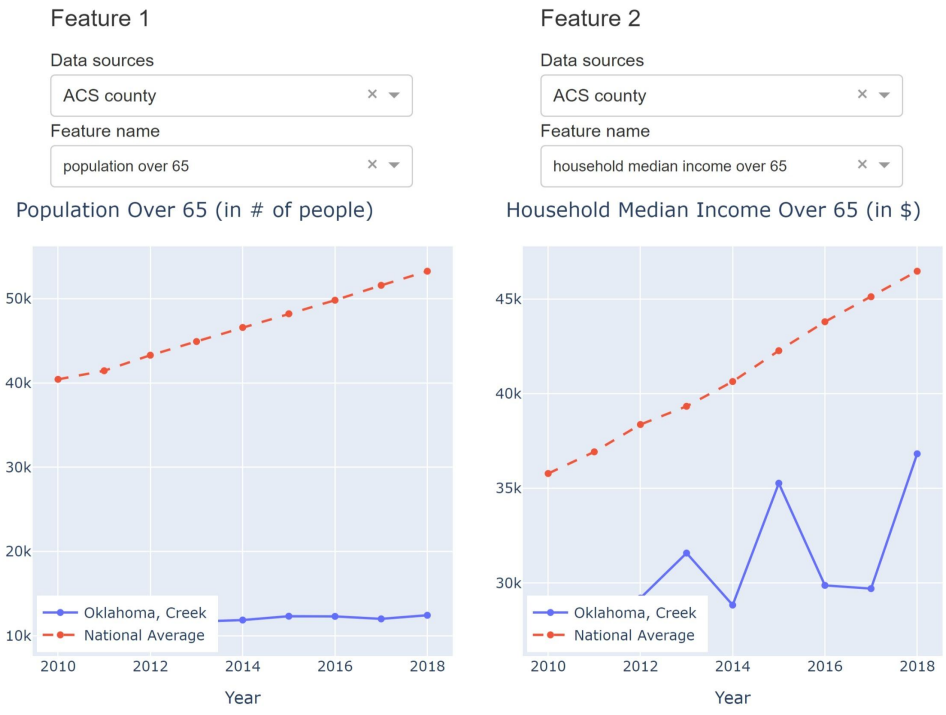


Figure 17: State Features Trend