Healthcare Analytics: Summer Project

code file-” summerprojectipynb ”
Rohit Janbandhu 2020CS10375
Aditya Goel 2020CS10317

Predicting LOS without using patient disposition

- Steps we performed-

  1. Access source data resident in a SQL database.
  2. Read data into Python Pandas Dataframes.
  3. Conduct data cleansing.
  4. Using target encoding for categorical variables (e.g. age group, race, etc.)
  5. Split data into a holdout set and conduct 10-fold cross-validation (CV) on the remaining data.
  6. Determine relevant model parameters.
  7. Evaluate the performance of the selected models.

- Output

Regression Model	10 Fold CV R2 score	Test Score R2 score
Random Forest	0.41367267918340944	0.416678985614406
Decision Tree	0.3985329684998221	0.3963520858708752
XGBoost	0.4133976819160945	0.4176664009898794

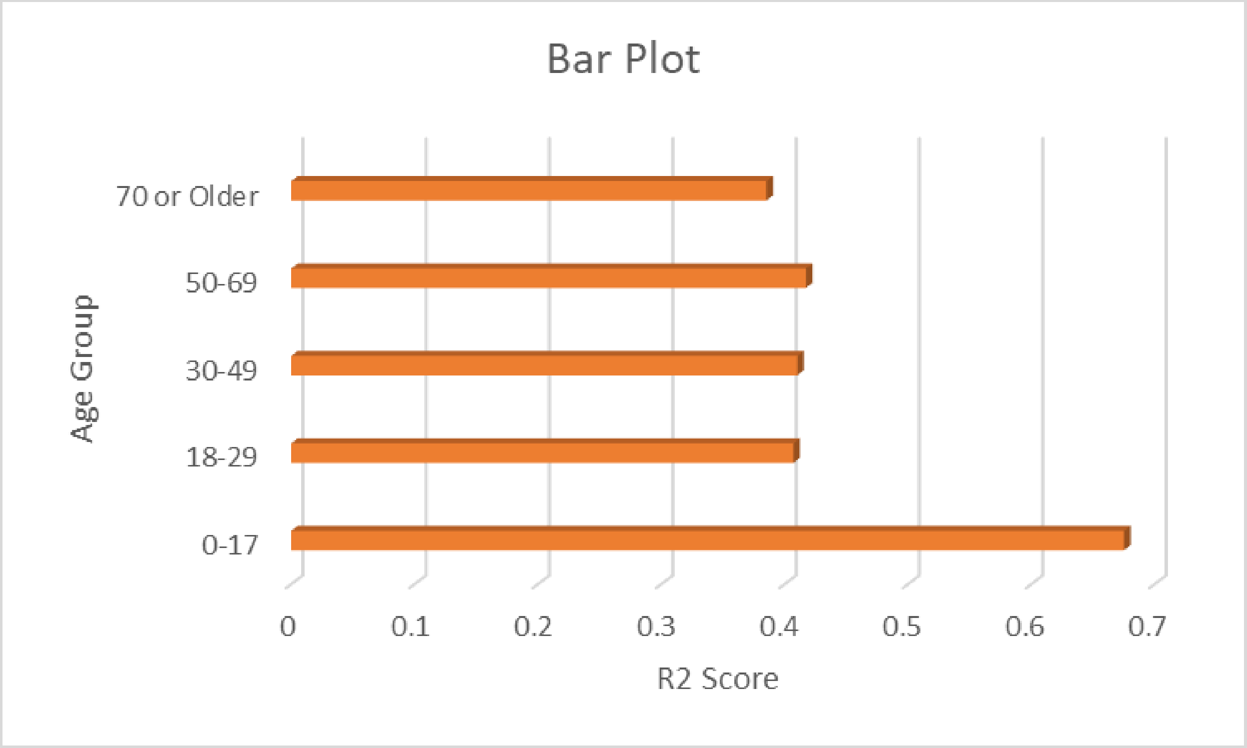
- Conclusion
  After removing patient disposition, The best regression model was XGBoost, with an R^2 value of .417.We see that the performance of the models dropped very little after removing the patient disposition feature.R2 dropped from .46 to .41. It is valid as we have earlier seen that the patient disposition feature is not very important in determining Length of stay. The most important qualities we identified were the diagnostic-related group and the severity of the illness code (FROM SHAP PLOT).

Age-wise data segmentation

- Steps we performed-

    1. Access source data resident in a SQL database.
    2. Read data into Python Pandas Dataframes.
    3. Conduct data cleansing.
    4. Using target encoding for categorical variables except for age group (e.g. gender, race, etc.)
    5. Splitting the dataset into different age groups.
    6. We split data into a holdout set for each age group and conducted 10-fold cross-validation (CV) on the remaining data.
    7. Determine relevant model parameters.
    8. Evaluate the performance of the selected models.

- Output
  Between all the regressor models, the Catboost Regressor shows the best result between all the regressor models. So only displaying the result from catboost regressor



Age Group	Fraction of Dataset	R2 Score	Percentage of Dataset
0-17	0.137291	0.675203	13.729100
18-29	0.091505	0.407089	9.150525
30-49	0.188470	0.410303	18.847007
50-69	0.283867	0.417400	28.386735
70 or Older	0.298866	0.385347	29.886634

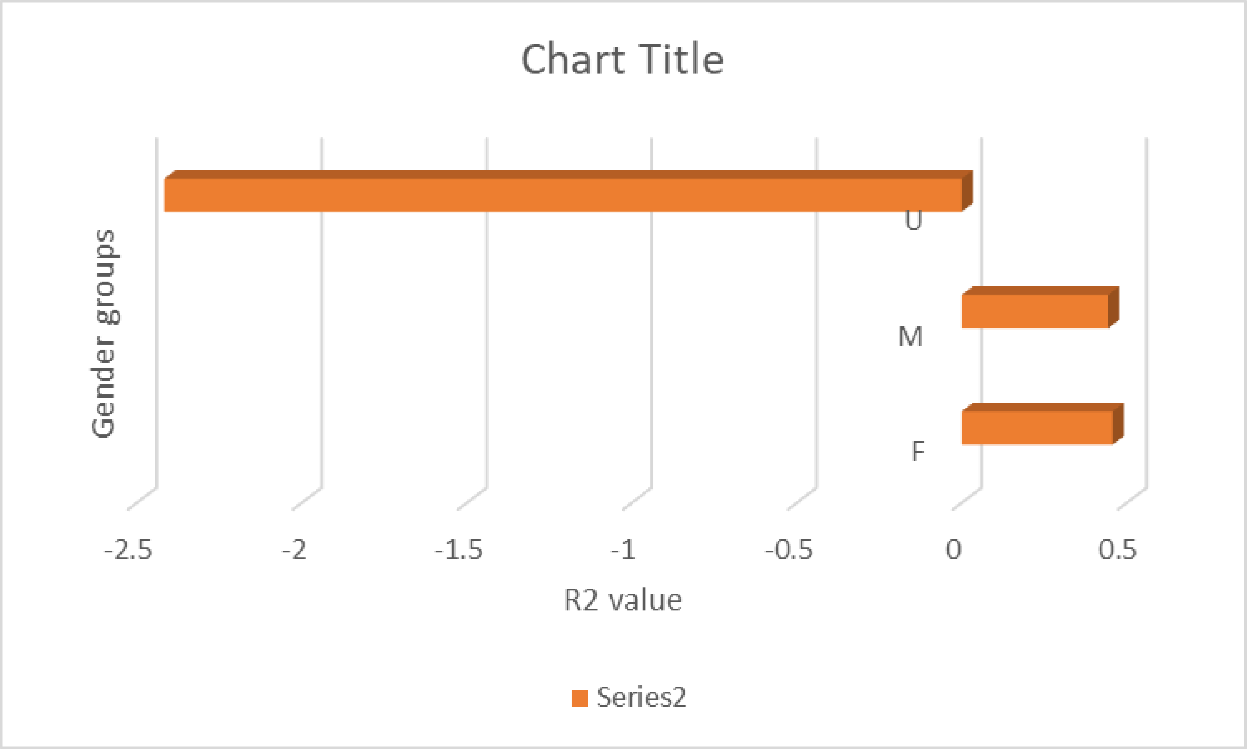
- Conclusion

  For age-wise data segmentation. We found out the best-performing model is the catboost regressor. Also, in the different age groups, for age groups 0-17, we are getting the best R2 value. Reason- Maybe Our model predicts this group better because the standard deviation in the length of stay values in this group is slight small, [redacted]indicating data are clustered tightly around the mean. Other groups have a high standard deviation, indicating that indicators data are more spread out.

Gender-wise data segmentation

- Steps we performed-
    1. Access source data resident in a SQL database.
    2. Read data into Python Pandas Dataframes.
    3. Conduct data cleansing.
    4. Using target encoding for categorical variables except for gender-group (e.g. age group, race, etc.)
    5. Splitting the dataset into different gender groups.
    6. We split data into a holdout set for each age group and conducted 10-fold cross-validation (CV) on the remaining data.
    7. Determine relevant model parameters.
    8. Evaluate the performance of the selected models.

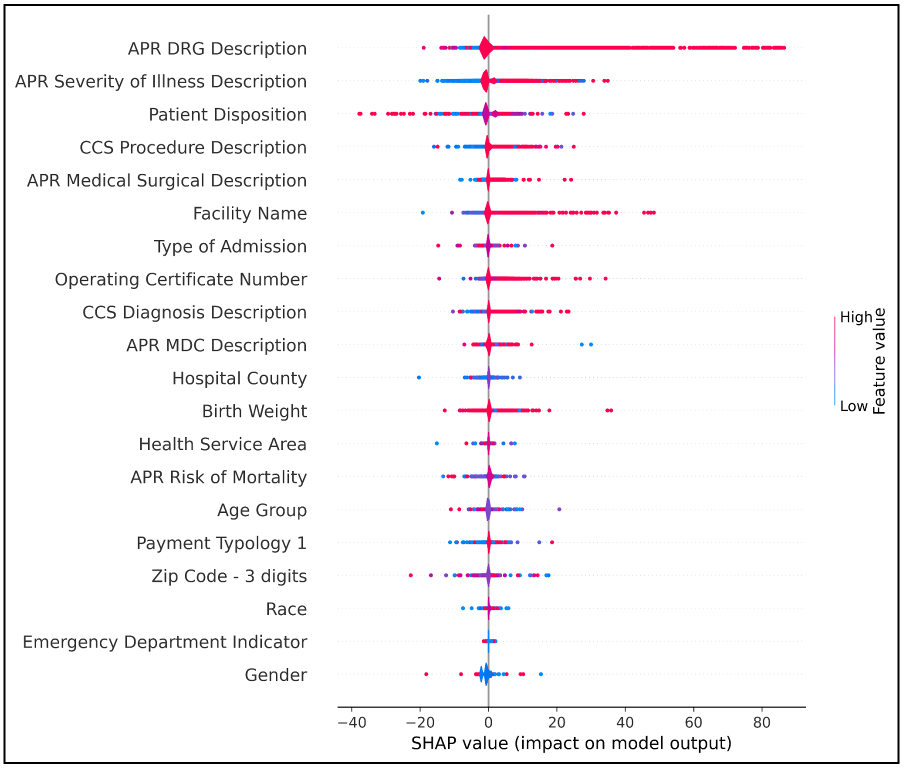
- Output
  Between all the regressor models: We found that different models predict different r2 values for the different gender groups. So we are only outputting the resulf of catboost regressor.



Gender Groups	Fraction of Dataset	R2 Score	Percentage of dataset
F	0.547549	0.457277	54.754938
M	0.452433	0.443529	45.243275
U	0.000018	-2.416728	0.001787

- Conclusion

  Between all the regressor models for different gender group we see different r2 values. But for different models, the maximum r2 value is of different groups. For the F and M groups, R2 values have increased to .45 and .44, respectively, from .41 . Also, for U, the R2 value we get is negative using a boost regressor. In the XGboost regressor, we get an R2 value of .44 for the U group. There are only 41 rows of data on it(Its percentage of data is almost negligible as compared to other groups). Reason- We have seen earlier In the feature importance graph that (SHAP plots) as gender as a feature have a very little influence in predicting LOS. Our model is performing better in the F group than the M group because of its higher percentage of datasets compared to the M group.

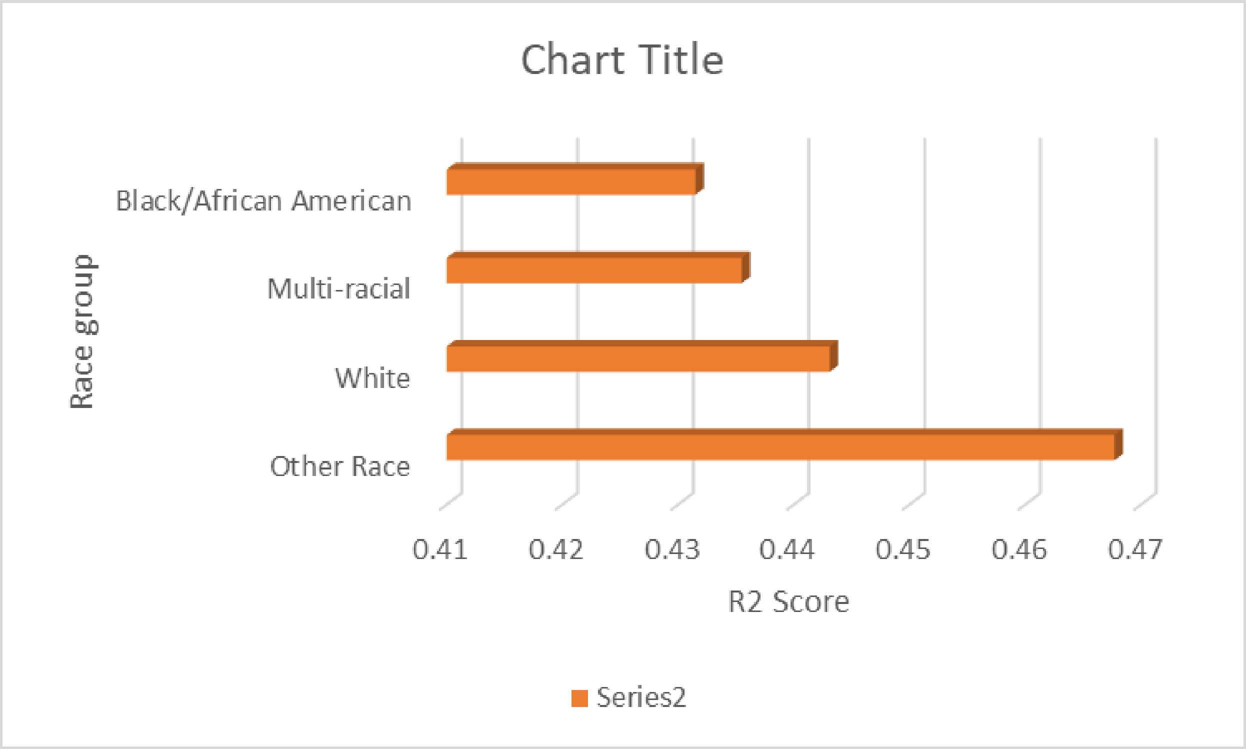


Race wise data segmentation

- Steps we performed-

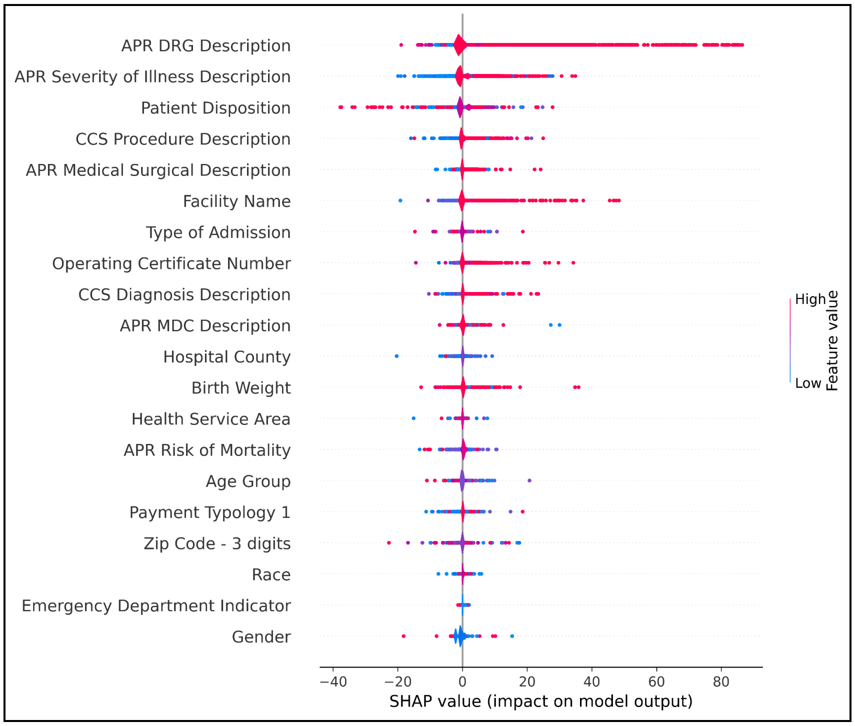
    1. Access source data resident in a SQL database.
    2. Read data into Python Pandas Dataframes.
    3. Conduct data cleansing.
    4. Using target encoding for categorical variables except for race (e.g. gender, age group, etc.).
    5. Splitting the dataset into different race groups.
    6. And for each age group, we split data into a holdout set and conducted 10-fold cross-validation (CV) on the remaining data.
    7. Determine relevant model parameters.
    8. Evaluate the performance of the selected models.

- Output
  Between all the regressor models: We found that different models predict different r2 values for the different race groups. So we are only outputting the result of the catboost regressor.



Race Groups	Fraction of Dataset	R2 Score	Percentage of Dataset
Other Race	0.246353	0.467711	24.635332
White	0.562687	0.443093	56.268695
Multi-racial	0.010304	0.435484	1.030423
Black/African American	0.180655	0.431483	18.065550

- Conclusion
  Between all the regressor models for different race groups, we see different r2 values. All the R2 values we get are in the range of .4 to .5. But for other models, the maximum r2 value is of other groups. Using a catboost regressor, the values of R2 are improved from .41 to .46. Reason- We have seen earlier
  In the feature importance graph, we have seen earlier that (SHAP plots) as race as a feature have very little influence in predicting LOS.



GeoSpatial analysis

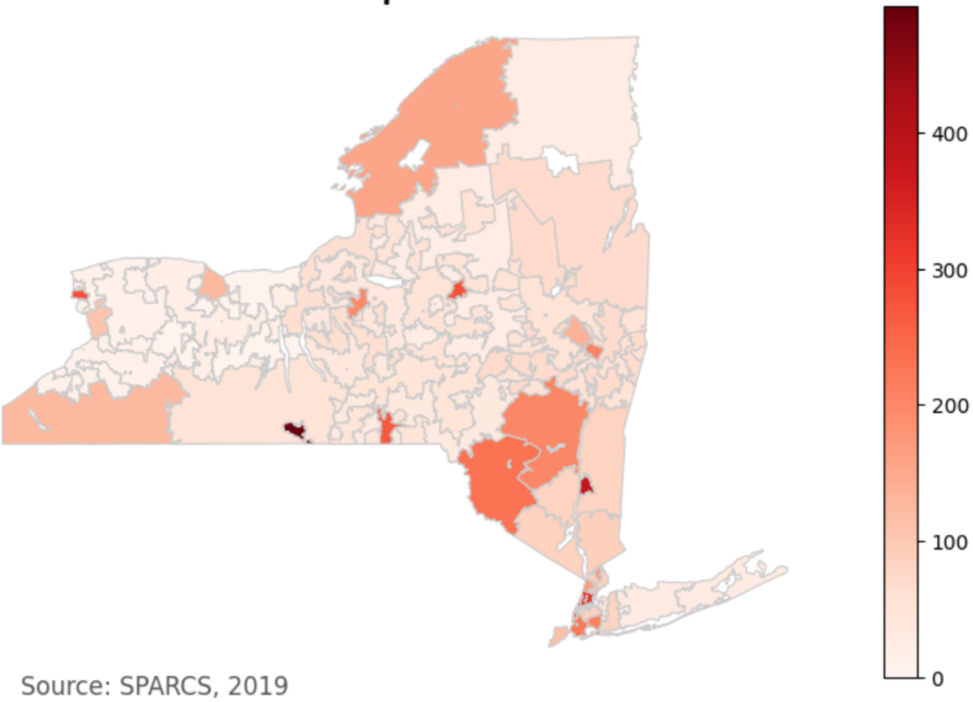
- Steps we performed-

1. Load the shapefile containing boundaries of zip codes (5 digits) in New York State as a geopandas data frame.
2. Load the CSV file containing the zip-code-wise population data of NYS in a pandas data frame.
3. Convert the datatype of the Zip Code column in the above data frames to int from object.
4. Merge the two data frames by conducting an inner join on the zip code columns of both data frames.
5. On the joined data frame, convert the five-digit zip into a three-digit one by first doing a type conversion from int to string and then taking the first three characters of each zip code.

6. Dissolve the polygons based on the 3-digit zip codes, aggregating the populations and removing all columns except those required, i.e., zip code, population and geometry.
7. Convert back zip codes from string to int data type.
8. Load the SPARCS data in a pandas dataframe.
9. Aggregate the data based on zip code and APR DRG Codes, maintaining the size of each group in a column titled “patient count”.
10. Filter to a specific disease code
11. Process the data again to remove any junk data, if present.
12. Convert the zip codes to int datatype.
13. Merge the data frames obtained in steps 7 and 12 by conducting an inner join on the zip code columns of both data frames.
14. Add a new column in the resultant merged data frame called 'patient_count per 100,000 people' obtained by dividing patient count in a zip code area by the population of that zip code area and multiplying the resulting number by 100,000.
15. Create a plot, add a title, create an annotation for the data source, Create colour as a legend, empty array for the data range, add the colour to the figure, and plot the figure.
16. Repeat the above steps for all major diseases in the dataset.

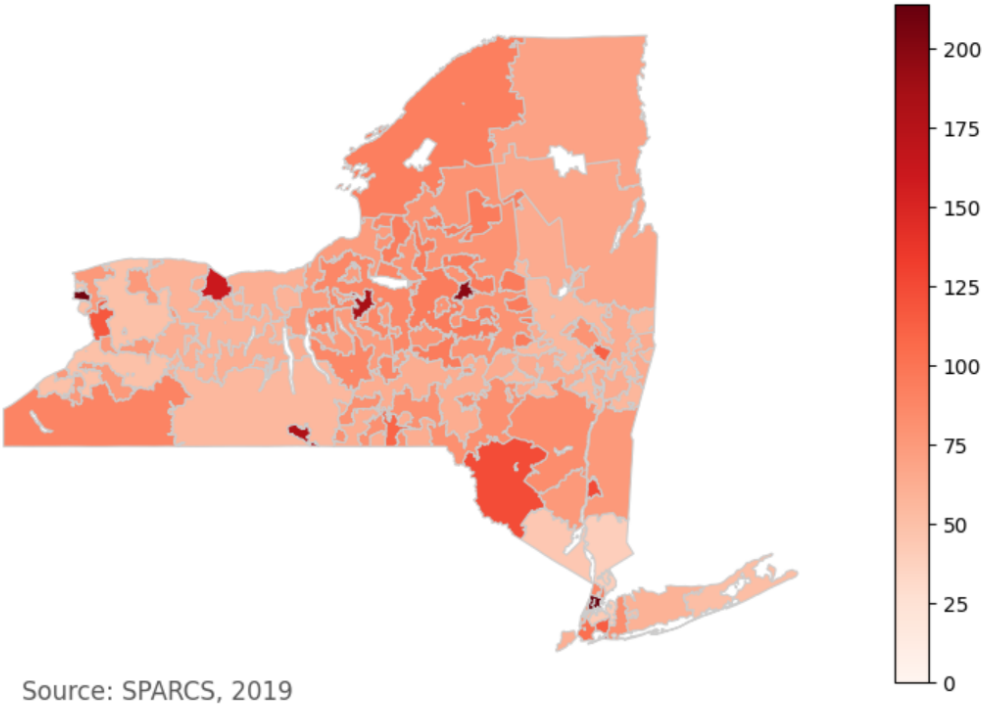
- Output

Schizophrenia



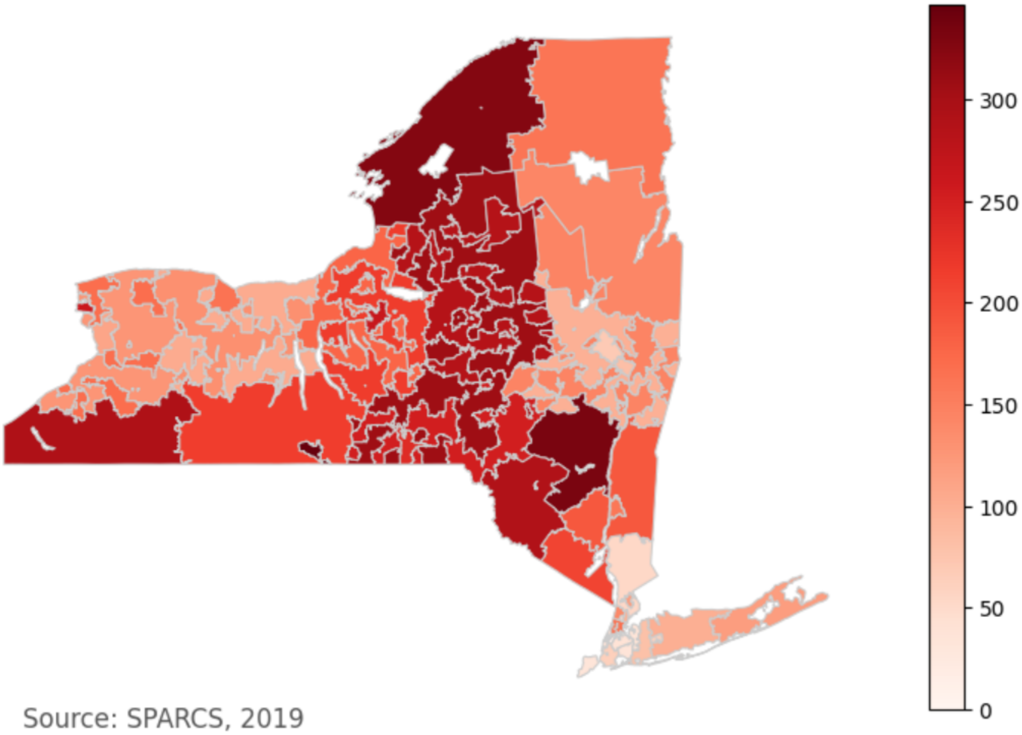
Source: SPARCS, 2019

DIABETES MELLITUS



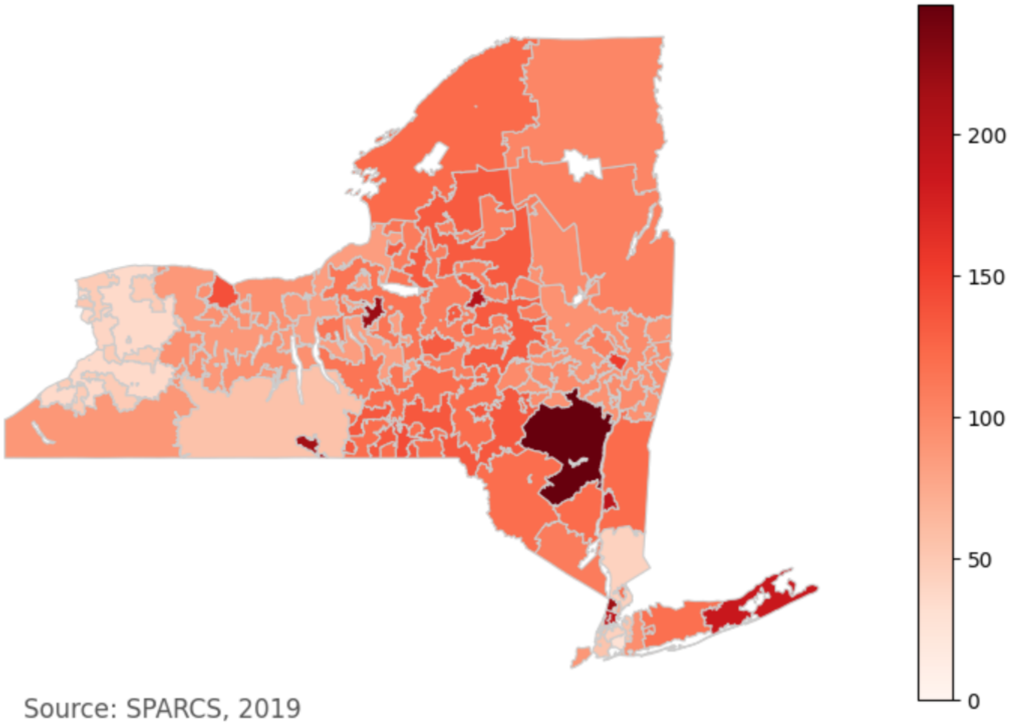
Source: SPARCS, 2019

Pneumonia



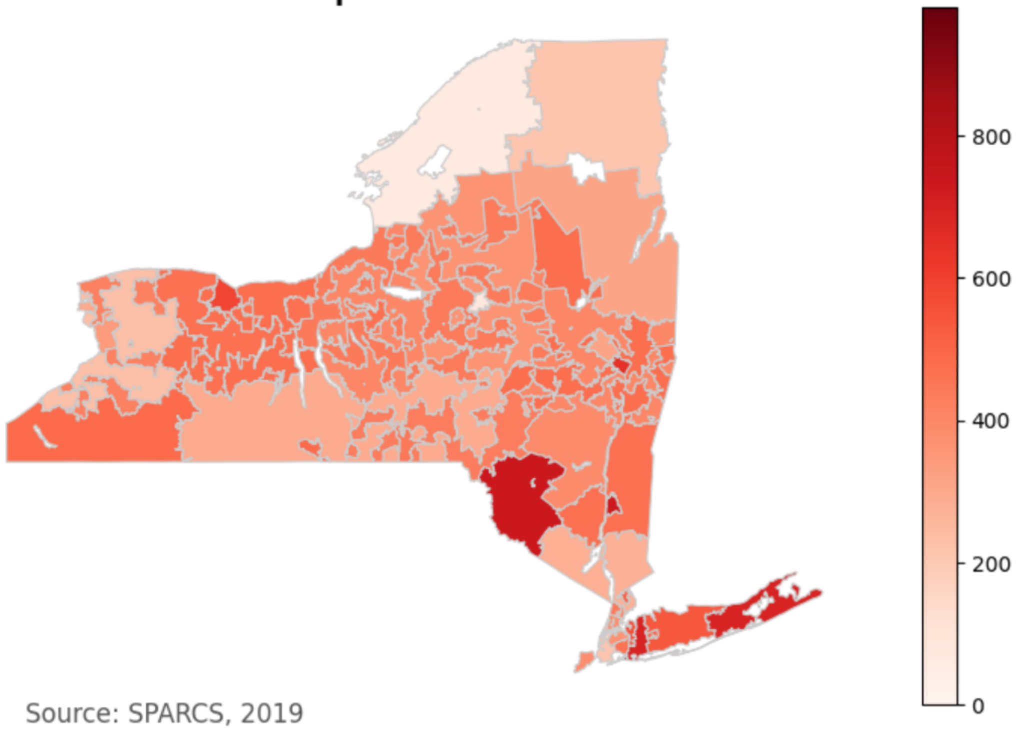
Source: SPARCS, 2019

CELLULITIS & OTHER SKIN INFECTIONS



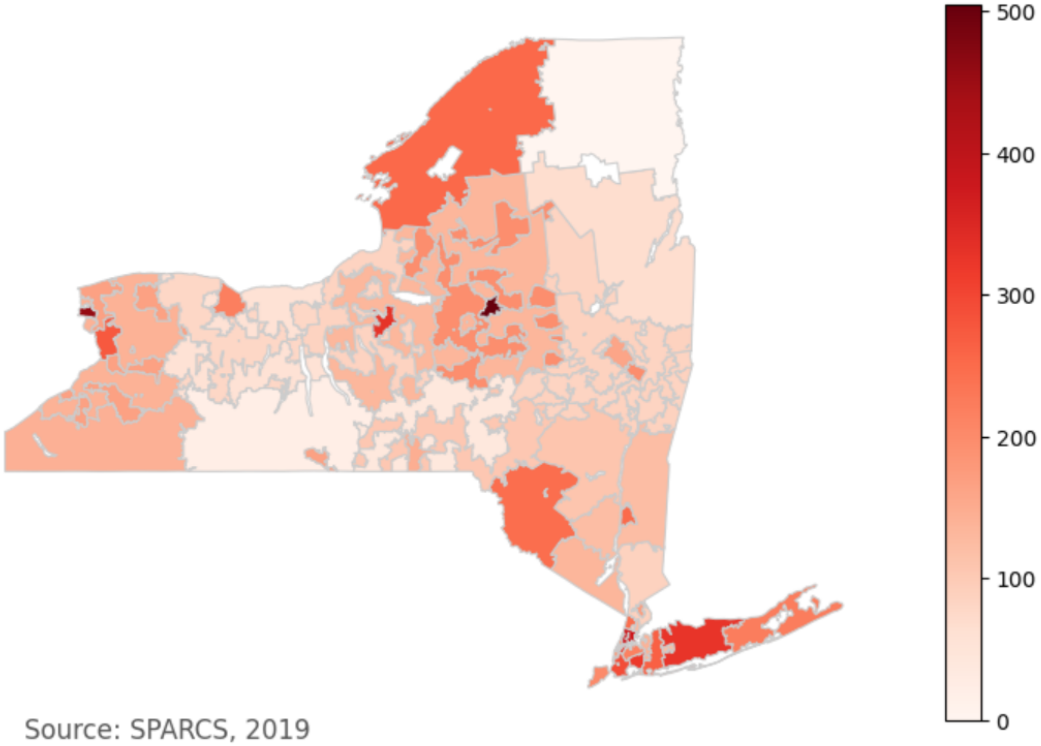
Source: SPARCS, 2019

Septicemia



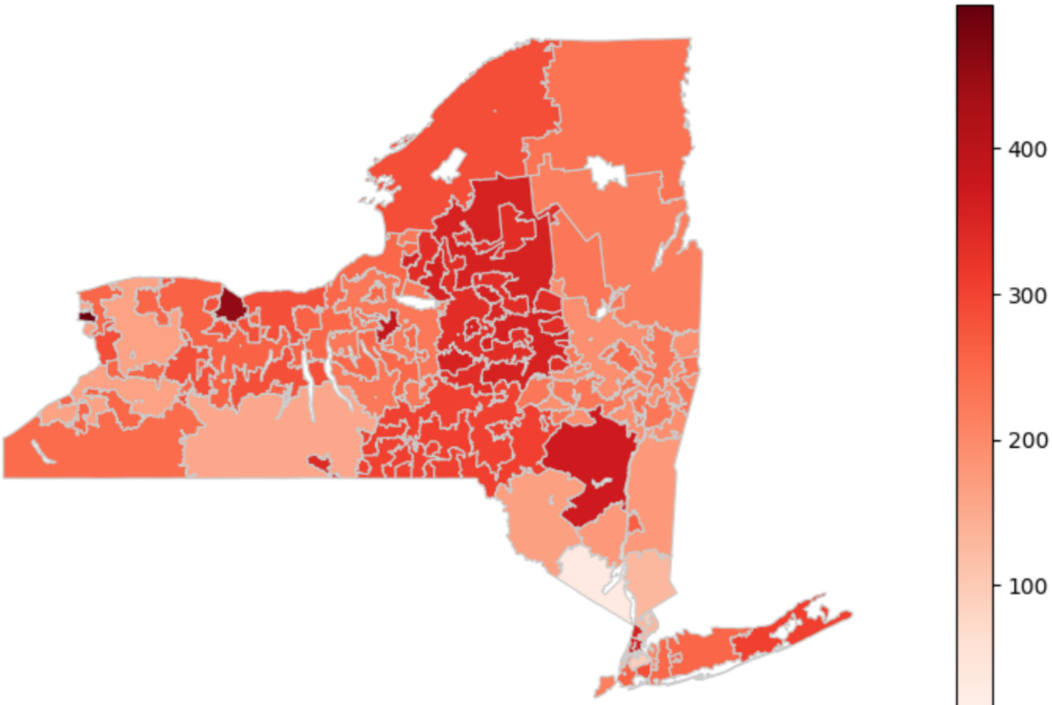
Source: SPARCS, 2019

CESAREAN DELIVERY



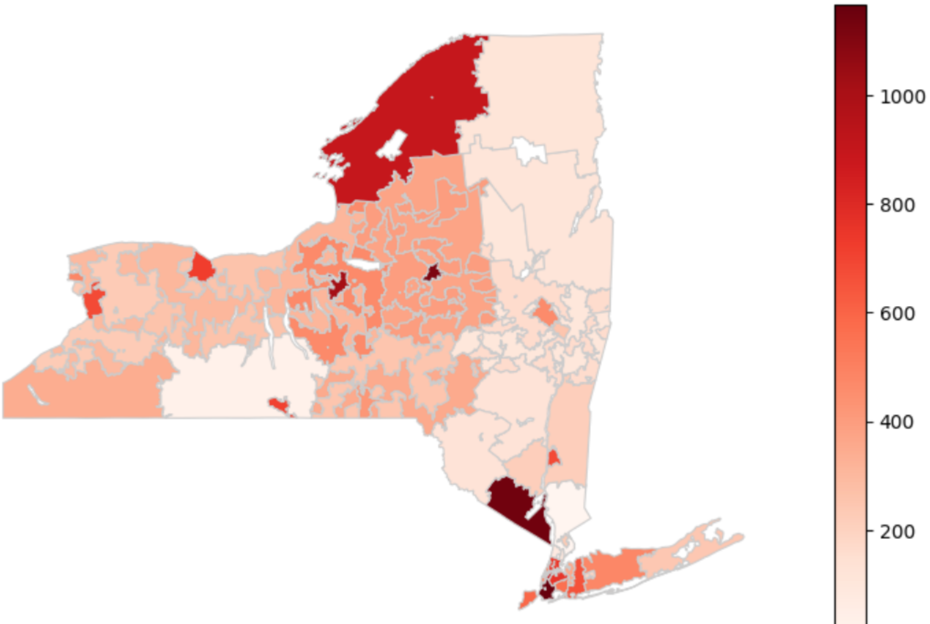
Source: SPARCS, 2019

Heart Failure



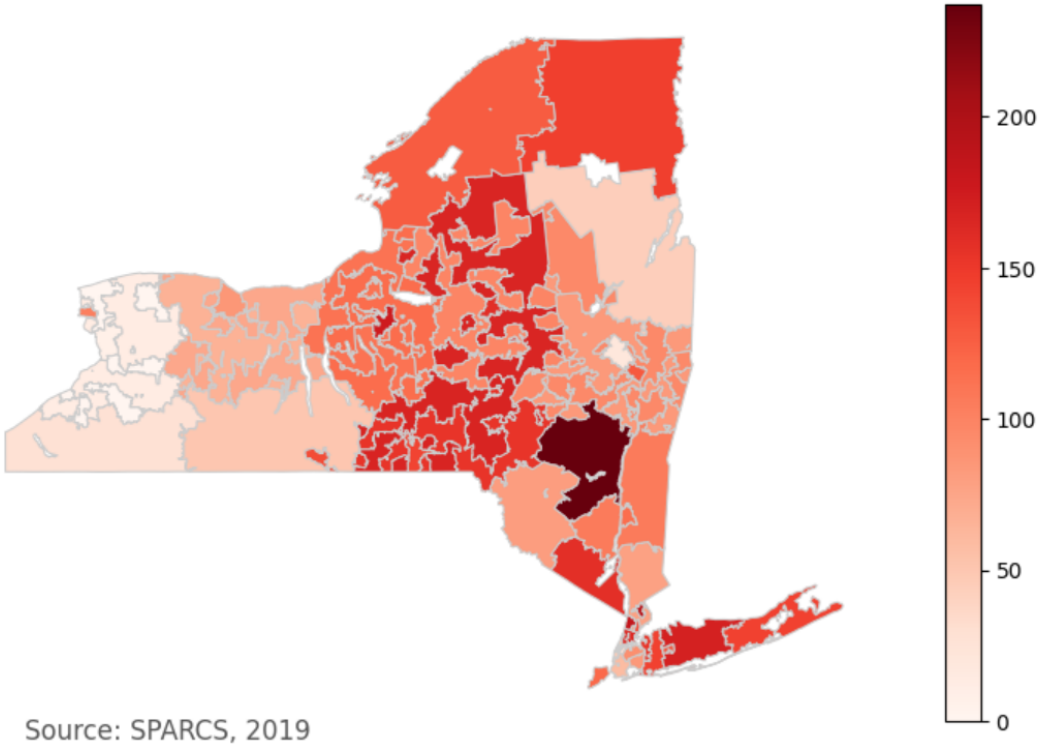
Source: SPARCS, 2019

Liveborn



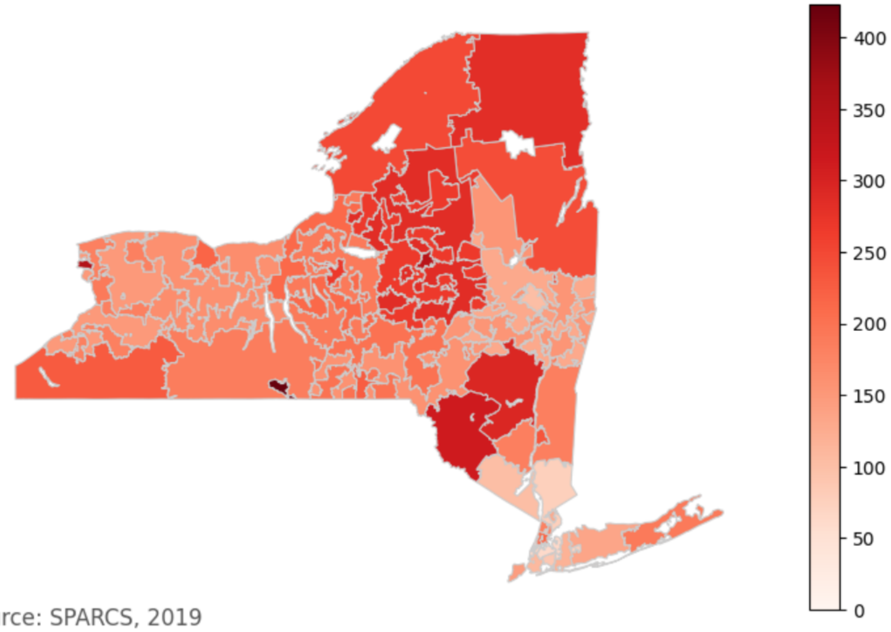
Source: SPARCS, 2019

URINARY TRACT INFECTIONS



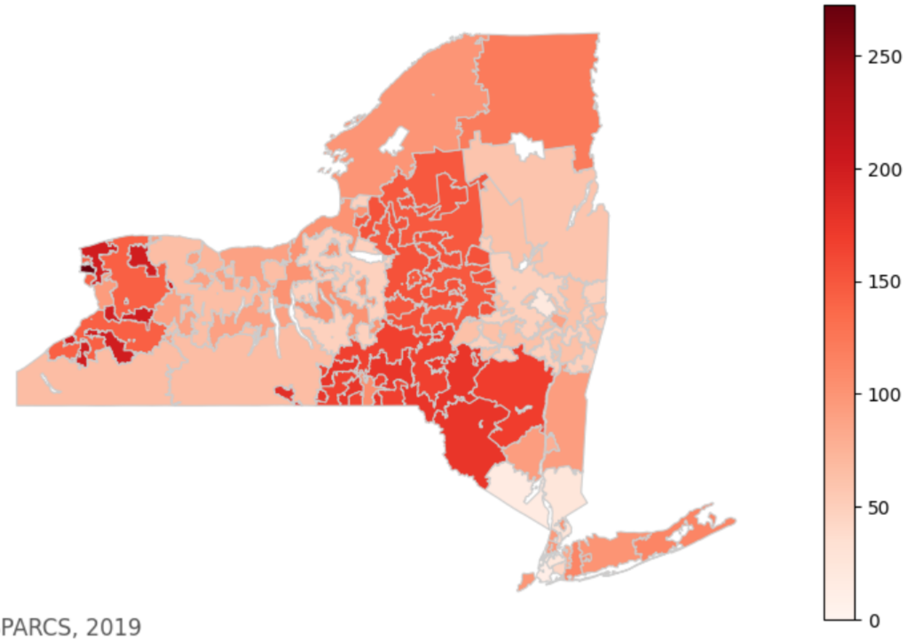
Source: SPARCS, 2019

CHRONIC OBSTRUCTIVE PULMONARY DISEASE



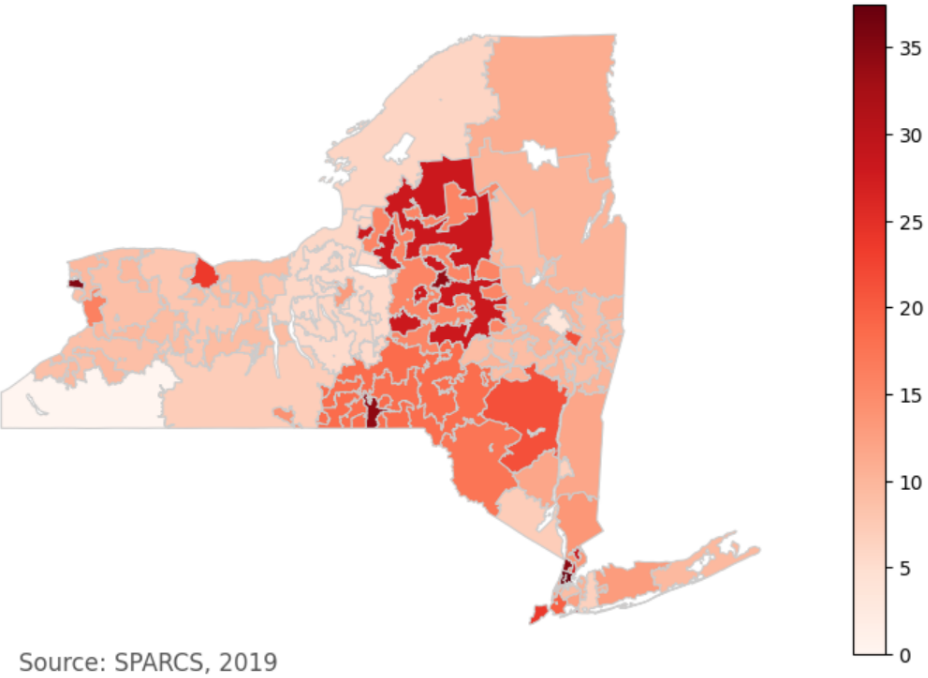
Source: SPARCS, 2019

CARDIAC ARRHYTHMIA & CONDUCTION DISORDERS



Source: SPARCS, 2019

Breast Cancer



Source: SPARCS, 2019

- Conclusion

  Different factors impact the geographical spread of different diseases. Factors like temperature and demography are common determinants in the geographical spread of diseases. The graphs obtained above are in line with our expectations, we don’t see any abnormal behaviour here. With this data, we can add other datasets like population and climate data of each of the counties to further continue this research work.