World Health Organization - Life Expectancy
Michael Tapia
Data 1030: Hands-on Data Science - Fall 2020 - Brown University
Github Link:https://github.com/michael-tap-ia/FinalProject_D1030

Introduction

The World Health Organization maintains the Global Health Observatory data repository, which contains various demographic information that is used to prepare research on health trends. The data is well documented, and some previous projects investigated the contributing factors of life expectancy, including general positive correlations with immunizations and increased life expectancy. I look to use this data, composed of 2,938 observations and 21 features, excluding the target variable, to understand the following questions surrounding life expectancy:

- What are the social/demographic factors that have positive or negative influence?
- What are the medical factors that have positive or negative influence?
- How can countries improve their life expectancy?
- What is life expectancy when presented with a new set of data?

It is important for countries to identify the expectation of life, specifically for relief funding and humanitarian efforts. If we can predict it from contributing factors, we can distinguish when and where relief efforts or policies are needed and at what volume. I look to know which factors hold the most influence over life expectancy to predict the life expectancy for incoming feature data from new years, if applicable. This regression problem will be addressed through the use of a XGBoost Regressor Algorithm model.

Exploratory Data Analysis

The data depicts 193 countries' demographic information from the years 2000 to 2015. A certain group of countries have much less records than others. We can see in Figure 1, there were 10 countries that only had records for the year of 2013 and no other year. As we will see later, these counties also do not have target variables to provide the model and as a result these ten records will be dropped to improve our overall model.

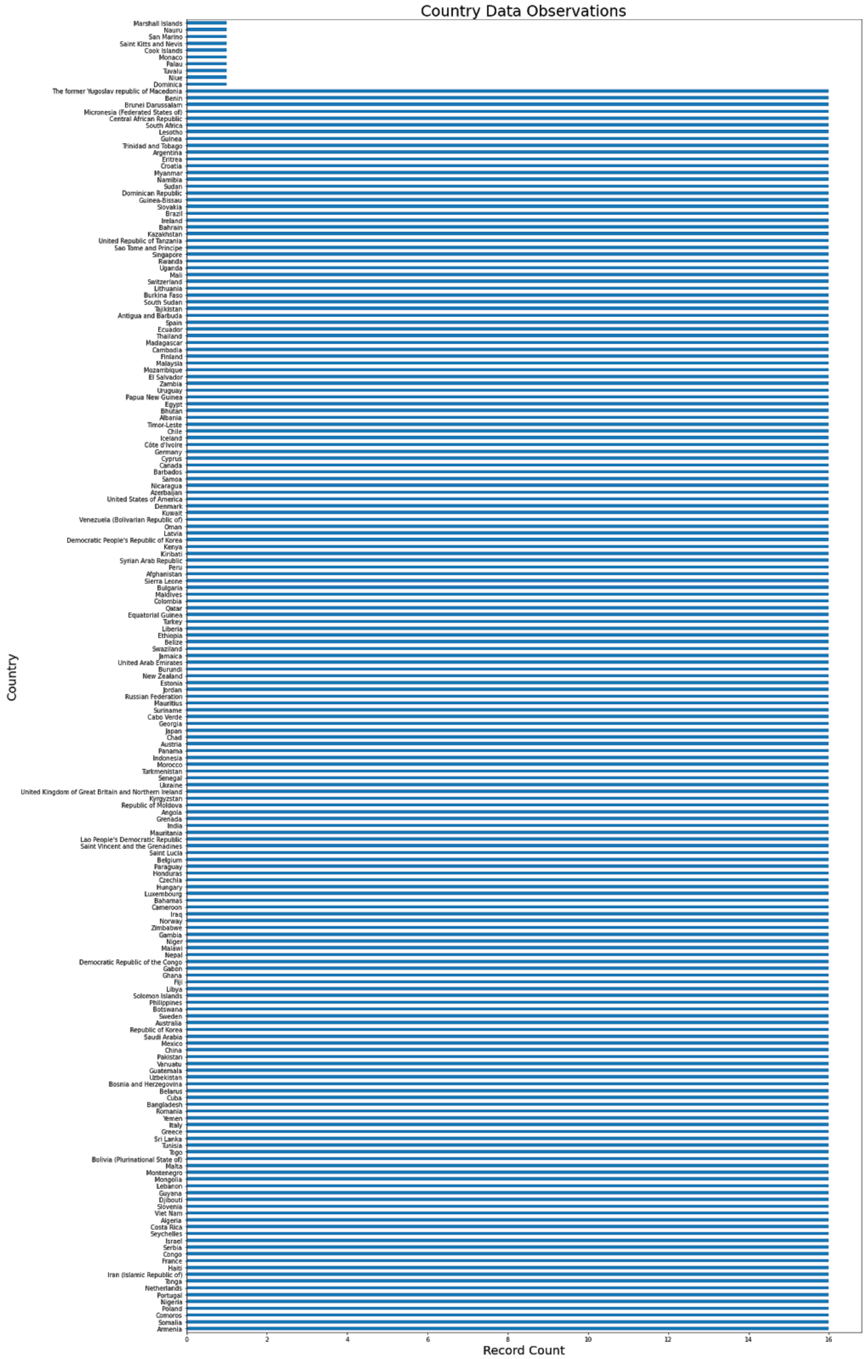

Figure 1: Bar chart of count of records by countries (193 unique countries)

When comparing the binary country status grouping in Figure 2, I find that a large sum of my information is composed of 83% developing countries and 13% developed countries. This would be of concern if we do not account for the context that our world has a larger amount of developing countries, but it is important to keep note that this could skew further down the line.

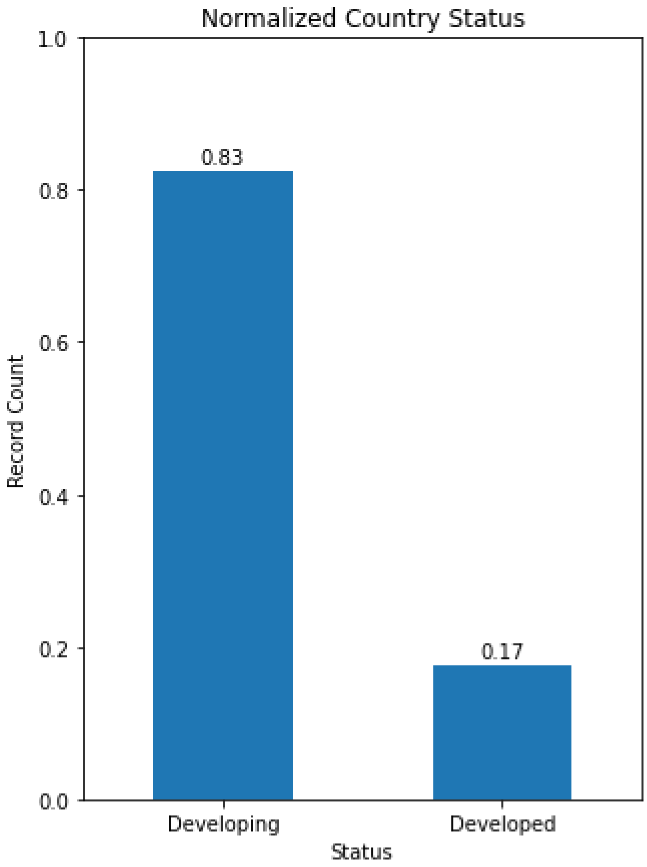

Figure 2: Bar chart displaying the percentage of records

One interesting observation as it relates to this feature, is that life expectancy is condensed higher in “age” in developed countries than developing ones. We look to Figure 3, to solidify our previous statement and observe that there is some overlap between “ages” 70 to 80, which closely resembles the life expectancy distribution peak right between these two “ages.”

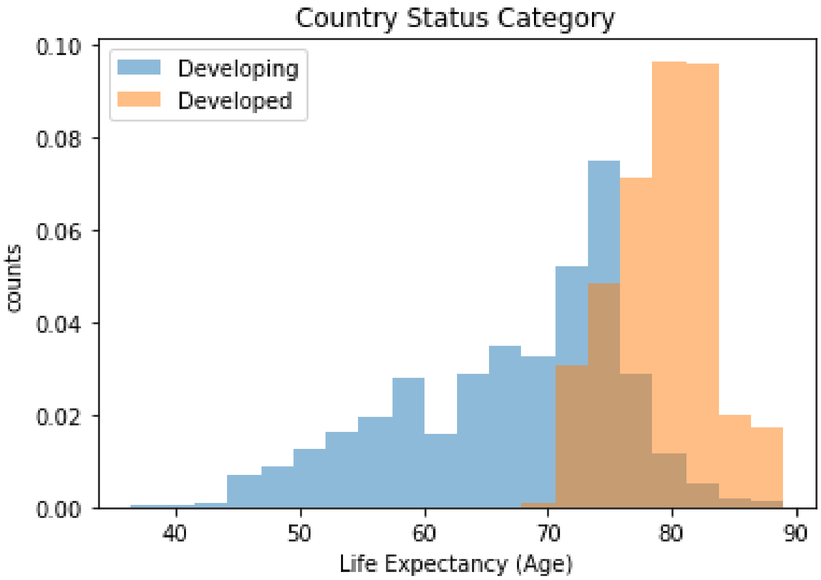

Figure 3: Stacked Histogram of life expectancy by country status

However, this data is laid out across 15 years so after creating multiple dual-feature plots to compare to my target variable, I was able to see 3 features with the strongest positive correlations to life expectancy over time. In figure 4, we see that "schooling", "income composition of resources," and "body mass index" have an overall impact, especially a positive one, as each of these variables increase, so does the target variable.

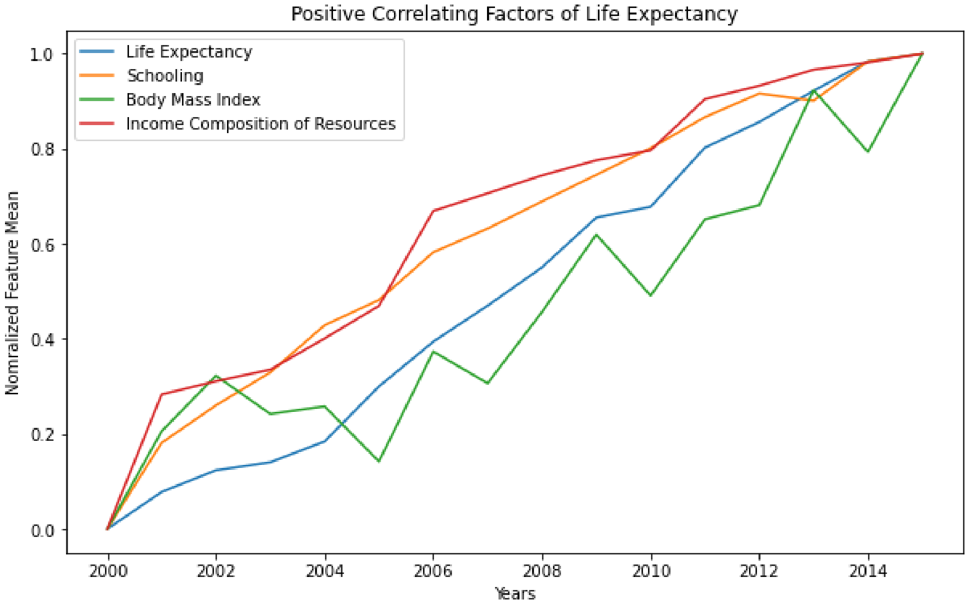

Figure 4: Multi-line Plot that displays the normalized mean of features that correlate to Life expectancy.

Interestingly enough, when we look even further into our dataset we can see that we have some correlation within our features. I realized that there were some features that are strongly correlated because they are a subset of a larger feature space. In other words, they are redundant. As a result, I came to the conclusion that I could drop “Thinness_5-9_Years” and “Infant_Deaths” because they lived within the larger “Thinness_1-19_Years” and “Death_Under_5” space, respectively.

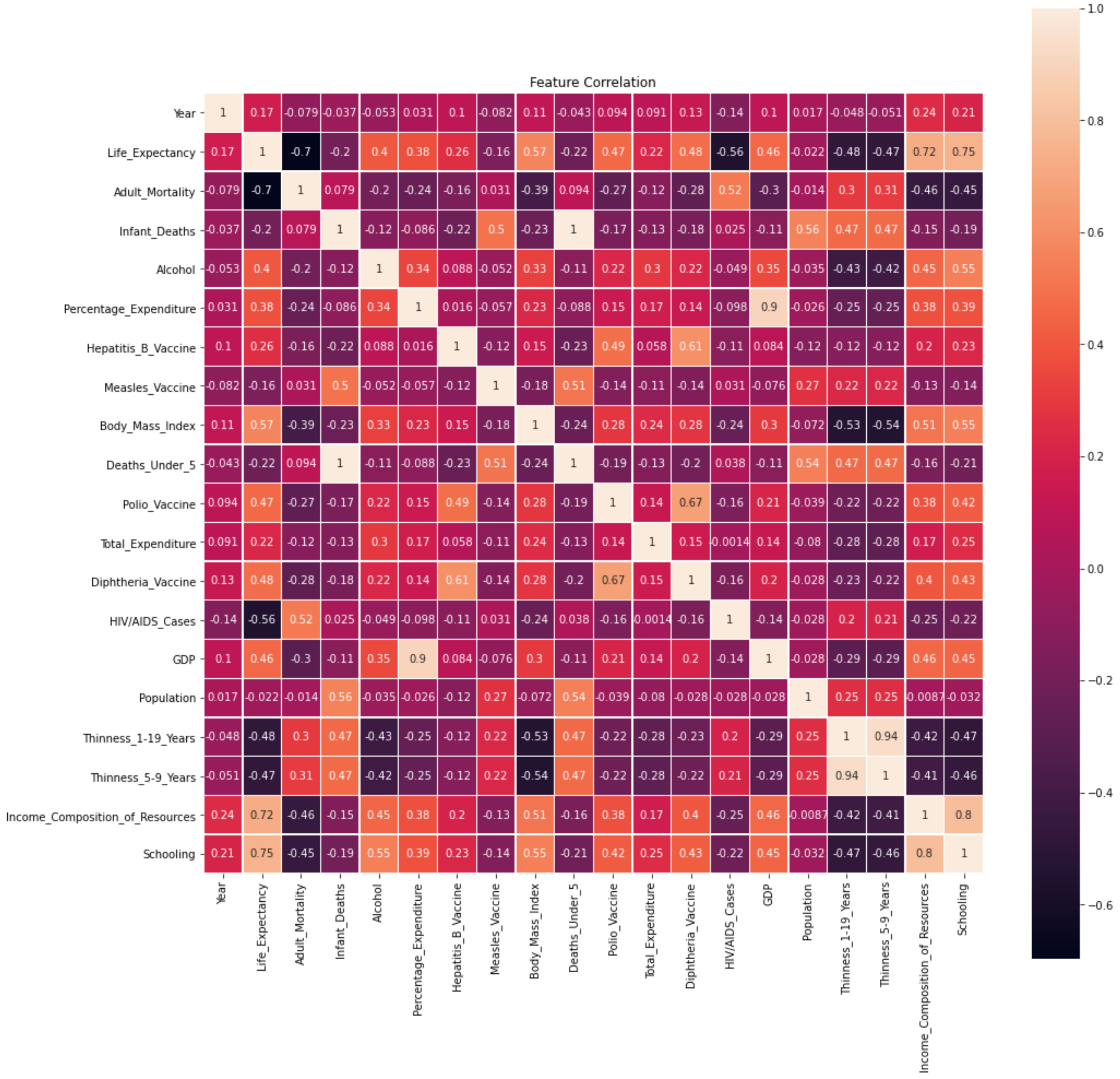

Figure 5:Correlation Heatmap between all feature and target variable

Methods

My non-IID data follows the countries' life expectancy in yearly intervals, therefore, new test data will come from data provided from countries in the future. I ended up making my splitting decision because of the way that time series splits the data to only include the oldest records and incrementally include newer data, which resembles how information would feed into the model in real life. However, to get my data ready for the split properly, and ultimately through the pipeline, I had to sort my data by the feature "Year." Next, I looked at how much of my data was missing. In figure 6, we see that there is a good percentage of missing values. After some experimentation, it would not be impossible to just drop all the rows with missing values as that leaves us with very little data to work with, with the exception that was stated earlier of the 10 missing target variable rows that will be dropped. All the remaining data points will use "IterativeImputer" to fill in "Missing At Random" and "Missing Complete At Random" null values, with the knowledge that with our data is sorted and iterating between non-null points will keep the linear trends already in place.

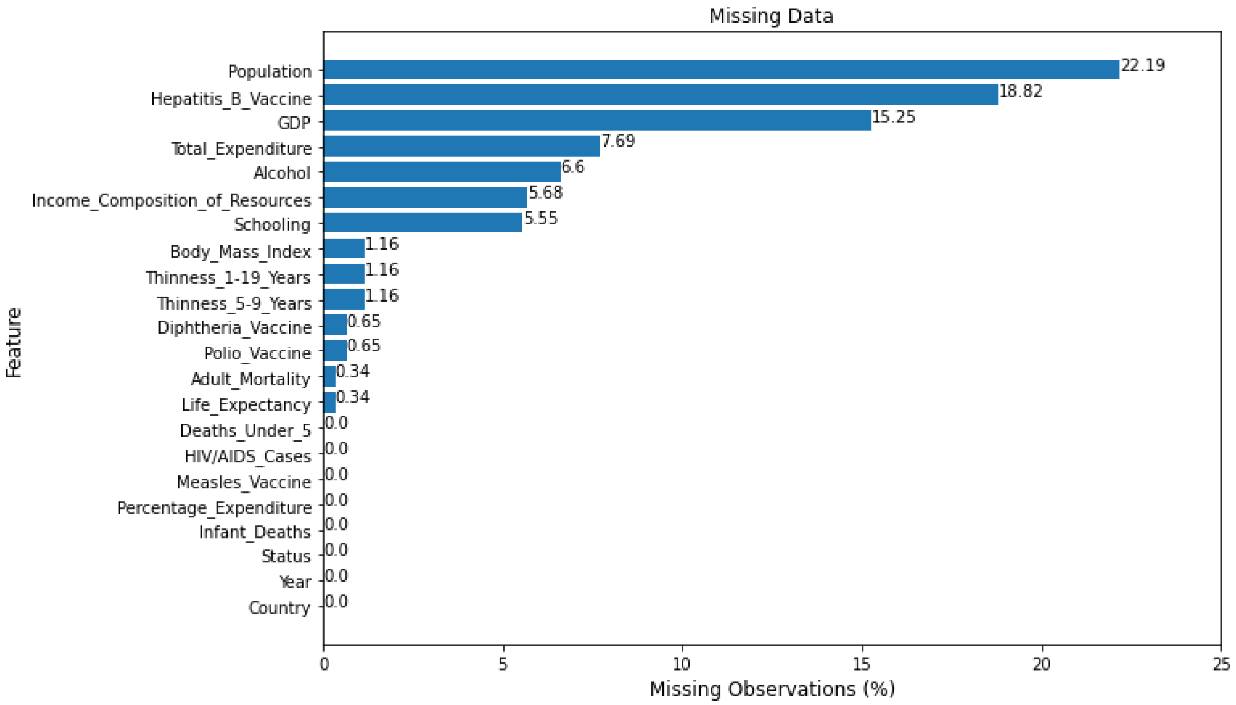

Figure 6:Bar plot of missing values (percentages)

Then I "OneHotEncoded" my rank-less categorical features and chose to use the safer option, StandardScaler, for my continuous features. I ended up with 201 preprocessed features that are split (1875, 201) for X_train, (468, 201) for X_validation, and (585, 201) for X_test. The splitting method used starts with the original sorted data splitting into 4

(TimeSeries) folds, where 3 folds contain the older “other” data for further splitting and 1 fold will be set aside as the younger, unseen data for test. I continue with my “other” data and conduct another time series split, where I take only the last iteration split as before and end up with 3 folds going to train and 1 fold to validation. Now, I have my data segmented as follow by date range:

Train < Validation < Test

With my data split out, we can see the result of my run thorough 7 different algorithms in figure 7 and 8, where I use the root squared mean error (where the smaller their error the better the fit) and R2 score (where the closer it is to 1 the better the prediction result) to evaluate the performance results, given that my target variable (Life Expectancy, which is an numerical age) is continuous. The first 3 algorithms are different linear regressions, where different alphas were set to tune. Next, the Random Forest algorithm was set to tune across different max features and max depth options. SVR has different C and gamma combinations. Knieghbors tune through different n_nieghbors and weights. Finally, XGboost tunes through different learning rates, colsample_bytree, and subsample combinations. The final results came back and the winner was the XGboost Regressor.

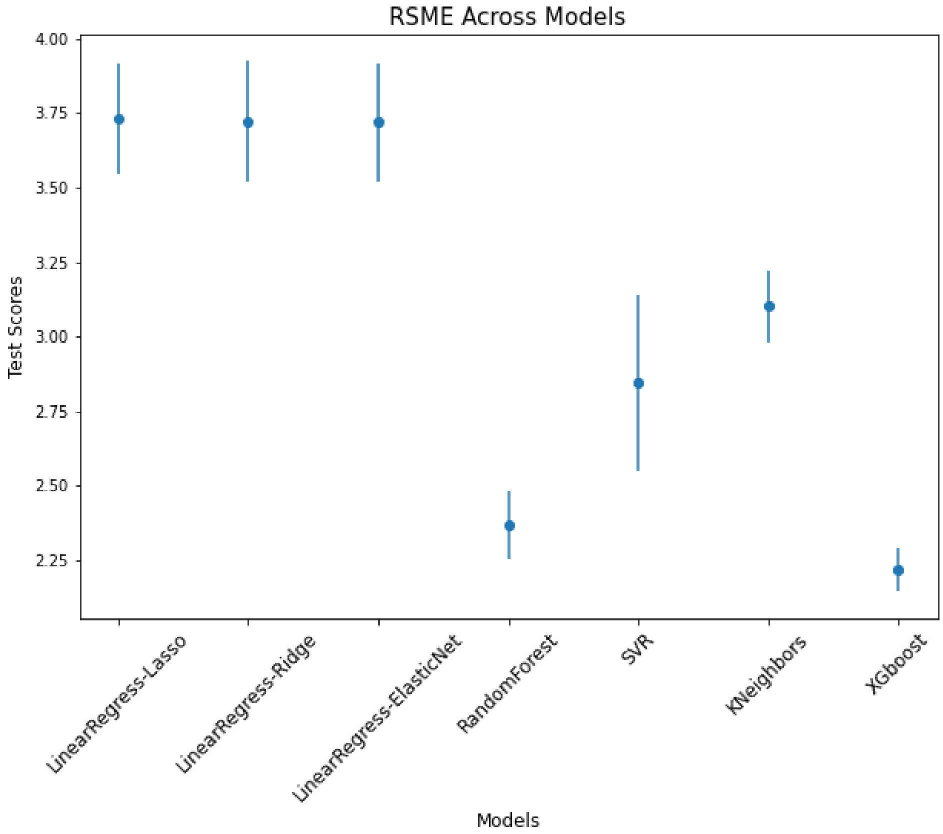

Figure 7:RSME across 7 regressors, with XGboost in 1st and Random Forest in 2nd.

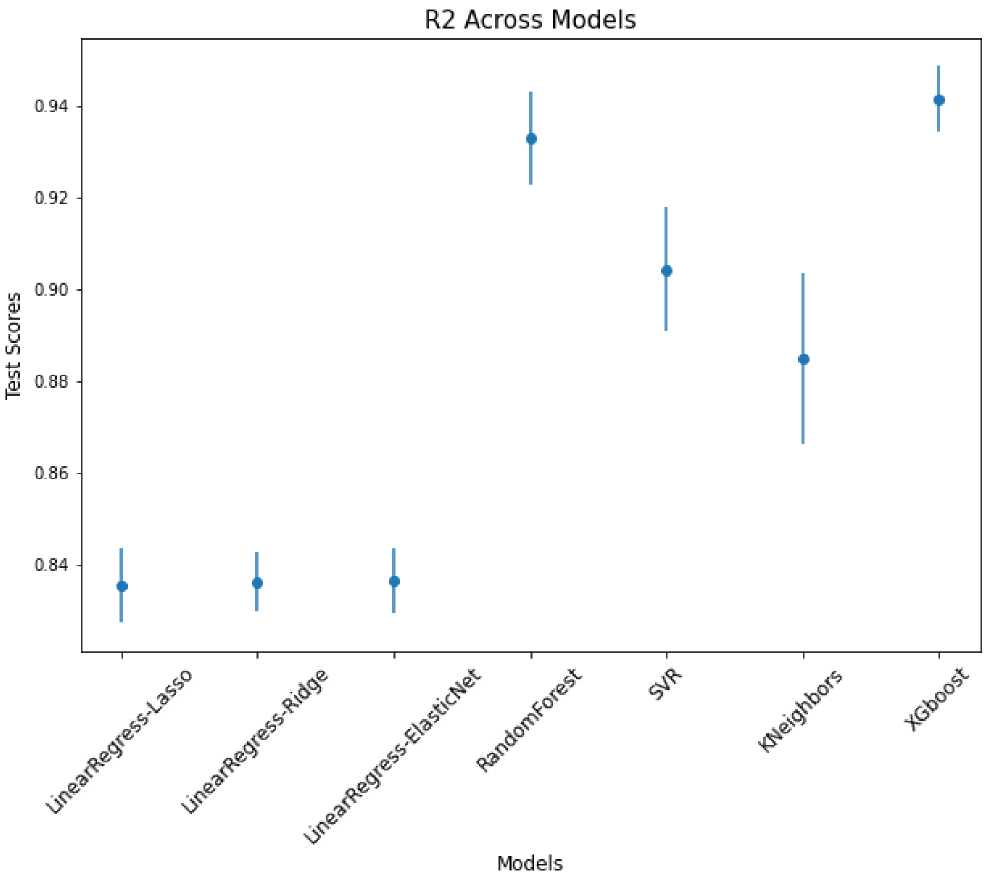

Figure 8: R2 across 7 regressors, with XGboost in 1st and Random Forest in 2nd.

Results

My final result for my Xgboost Regressor Algorithm TimeSeries Split model was better than the baseline with a R2 score of 92% and an RSME 2.37, where my baseline simple linear regression model score was 86% and a RSME 3.15. The features that influenced the model and answers the question presented in the first section of what factors impact life expectancy, can be seen below in figure 9.

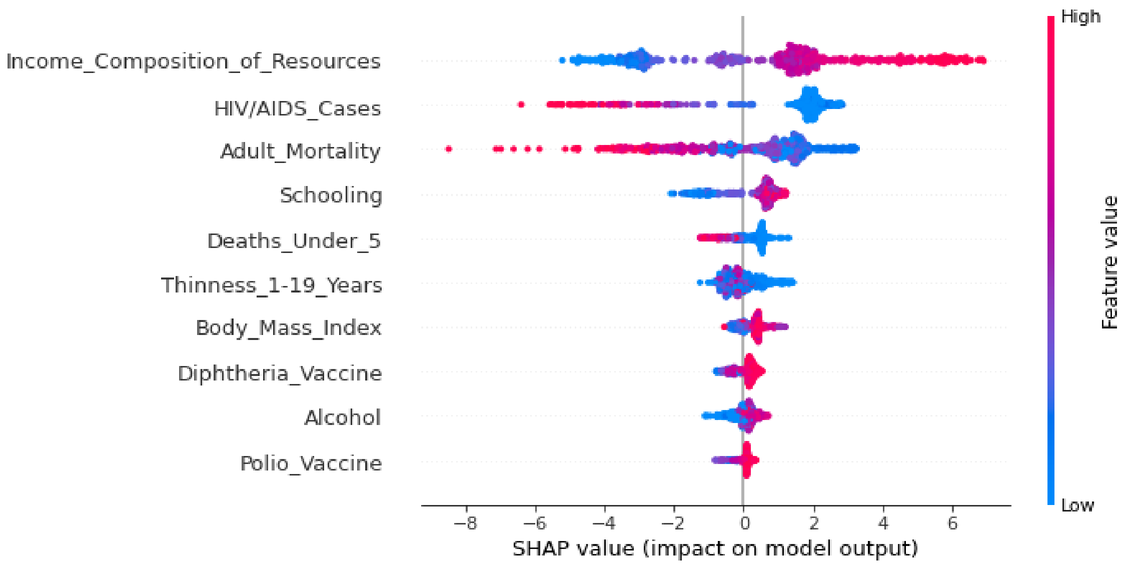

Figure 9:Global feature importance on model prediction

The top features to have an impact are income composition of resources, HIV/AIDS cases, adult morality, and average years schooling for the population. In context, it could be understood that if a country uses its resources productively, as defined by the Human Development Index, that we can expect life expectancy to increase. Adjacent to this factor, the more years of schooling a population has the better the life expectancy, however, there seems to be capitation of the amount of schooling one can get before the impact of life is minimal.That does make sense, as there is a point when the knowledge to be self sufficient is accomplished. That only leaves the negative impacting factors of HIV/AIDS cases and adult morality. Given these two factors are pretty much death metrics, the more amounts under these features the lower our prediction target will be.

Outlook

My current model can take in features and produce the life expectancy age, but it cannot forecast future life expectancy. It is a useful model given it has access to current data to provide the target variable for which it is based on. I could be even more useful if I were to use historical data to produce a trend analysis and return future Life expectancy based on the current data I have already collected. This would allow me to use my previous target variables as a feature to predict future target variables.

Citation

1) Imputation - Interpolate function for time series data: https://towardsdatascience.com/how-to-interpolate-time-series-data-in-apache-spark-and-python-pandas-part-1-pandas-cff54d76a2ea
2) Code Help from previous notebook on dataset: https://www.kaggle.com/stefanost/regression-with-thousands-of-missing-values
3) TimeSeires split: https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.TimeSeriesSplit.html
4) Outlook - Autoregressive: https://www.iunera.com/kraken/big-data-science-intelligence/time-series-and-analytics/top-5-common-time-series-forecasting-algorithms/