DATA 698 Final Research Project

Magnus Skonberg

2021-05-16

Contents

Abstract

With a high rate of avoidable deaths and chronic disease as well as an obesity rate at two times higher than the OECD average, the forecast for American health is gloomy. With that in mind, the purpose of this project was ultimately to answer three questions: 1) What United States counties are most favorable for an active, healthy lifestyle? 2) What are the differentiating characteristics that make them so? and 3) What might the best regression model be for modeling the relationship between our healthy lifestyle metric and these differentiating characteristics?

Our methodology for addressing these questions consisted of: Data Gathering & Preprocessing, Data Exploration & Preparation, and Multivariate Regression Analysis. We created our dependent health score variable, identified the top 10 healthiest counties, extended assumptions, gathered independent variables and brought all data into a consistent format. We then operated upon our dataset, identified our strongest and weakest predictors, and built our model with 10 of the original 24 independent variables.

With raw and transformed data, we then optimized on the coefficient of determination (R-squared), explored numerous linear regression vs. regularization models, output performance metrics in tabular form, and selected the best regression model for modeling the relationship between healthy lifestyle and our independent variables. We used the R-squared and RMSE values as part of our selection criterion and found the linear regression model applied to transformed data to be our strongest model, largely due to its strong predictive performance for unseen data (R-squared of 0.71).

Key Words: ***Feature Creation, Feature Selection, County Level Health, Regression, Regularization***

Introduction

The Author

Magnus Skonberg is a Consultant and Master of Data Science candidate at CUNY SPS, who resides in Middle Island, NY. Although this was a solo project, I (Magnus) would like to give a special thanks to Jeff Nieman, Avraham Adler, Jaan Altosaar, and Christian Thieme for their guidance.

The Problem

The U.S. spends more on health care than any other OECD county yet it has the lowest life expectancy, highest number of hospitalizations from preventable causes, highest rate of avoidable deaths and a chronic disease burden and obesity rate at two times higher than the OECD average.

The forecast for American health is gloomy, yet identifying the issue could aid in addressing it, and one way of doing so would be to highlight "bright spots". To highlight the areas across the United States where the general populace are living long and living well, while exploring why these areas are able to succeed. Recognizing influential correlations and shedding light on the areas most conducive for leading a long and healthy life may provide example for emulation and answer the following:

1. What United States counties are most favorable for an active, healthy lifestyle?
2. What are the differentiating characteristics that make them so?
3. What might the best regression model be for modeling the relationship between our healthy lifestyle metric and these differentiating characteristics?

In an attempt to answer these questions we'll create our own healthy lifestyle metric, gather typical and atypical variables, explore their relationships, and then explore the efficacy of numerous regression models. The data will contain entries for all 3006 American counties as well as corresponding states, health scores, population, alcohol consumption and numerous features to be discussed during the Literature Review section.

Literature Review

For this project I focused my research and literature review around health outcome drivers and means of feature selection.

While the scope of this project was to explore related United States county data, I took a broad approach and considered international and higher-level data to best determine which variables might be fused to create the dependent variable and which variables might be best to consider as independent variables (typical and atypical):

Dependent Variable Compilation

The first group of researched variables dealt with our dependent variable and the variables (perceived as) most correlated with healthy lifestyle.

Bhardwaj, Amiri, Buchwald, and Amram identified social and environmental correlates of healthy aging and longevity. They found that demographic, environmental, and social factors all play a significant role in predicting an individual's likelihood to lead a healthy life and live until an older age. From their study, the importance of longevity as a factor in living a healthy lifestyle was reinforced, as were a number of factors we might consider as independent variables when building a regression model (ie. education, socioeconomic status).

Kaminsky, Lessler, Sharfstein, and Wallace, observed the effect of clustering based on sociodemographic context when considering county-level percentile health rankings vs. nationwide rankings. They considered smoking, motor vehicle crash deaths, and obesity as factors and found that clustering based on sociodemographic characteristics allowed for a better understanding of how other factors may shape the prevalence of health outcomes. From their study, the importance of obesity as a healthy lifestyle factor was reinforced, as was the importance of nuance when interpreting health outcome variables at a county level.

The final factor for consideration in the make-up of our dependent variable was physical activity. The motivation for its inclusion came from Azevedo, Hallal, Wells, and Victoria's research. Although their study was primarily focused on physical activity amongst adolescents, they considered the short term and long term effects of physical activity and found that "there is an indirect effect on all health benefits resulting from adult physical activity".

Our dependent variable was a fusion of longevity, obesity, and physical activity county-level data. These three are important factors that take short and long term health into consideration, and thus their amalgamation provides a strong representation of healthy lifestyle (as a metric) at a county-level.

Independent Environmental Variables

An, Li, and Jiang explored the impact of environmental determinants (via EQI) on physical inactivity among U.S. adults at the county-level. Although, environment and physical activity vary substantially across the United States, the study identified an inverse relationship between environmental quality and physical inactivity.

Holick and Wacker provided an in-depth glance into the incredible health benefits of sun exposure and increased vitamin D production. They cite an inverse association with latitude and many chronic illnesses including: cancer, autoimmune disease, diabetes, depression, seasonal affect disorder, hypertension, etc. and recommended increased sun exposure (without sun burn) and vitamin D fortification and supplementation as a natural remedy to many of these ailments.

From consideration of the environmental quality index (EQI) and yearly sunlight, we move to that of socialization and crowding by considering population density and the effect (positive or negative) it may play on the lifestyle of its local inhabitants. Sven Bremberg analyzed the role population density played on life expectancy and years lost, and found that mortality rates were consistently high in Finland, Norway and Sweden in less densely populated municipalities. We might extend, from this, the notion that (to a certain point), a higher population density is more favorable for a longer, healthier life.

Independent Health Variables

In A National Study of the Association Between Food Environments and County Level Health Outcomes, Ahern, Brown, and Dukas applied linear regression models to estimate the relationship between food availability and access variables (ie. per capita grocery stores) with health outcomes. Positive health outcomes were linked to more per capita full-service restaurants, grocery stores, fitness and recreation centers, and direct farm sales, while negative health outcomes were linked to more fast-food restaurants and convenience stores. As a general takeaway we might extend that access to nutrient-dense foods if favorable to healthier lifestyle whereas living in a "food desert" would favor the opposite.

In Smoking, alcohol consumption and mental health: Data from the Brazilian study of Cardiovascular Risks in Adolescents (ERICA) Ferreira, Jardim, Jardim, Sousa, and Rosa found that second hand smoke, smoking, and alcohol consumption are all adversely associated with psychological distress in the adolescent population. When we extend this from adolescent Brazilians to consider adult Americans, it will be interesting to observe the strength of correlation between bad habits at scale and a county's standing health-wise.

Independent Economic Variables

Islam, Leer, Teo, et al. found that among patients with coronary heart disease or stroke event, the prevalence of healthy lifestyle behaviors was relatively low regardless of whether the nation under consideration was high vs. medium vs. low income, but the behaviors were at their lowest levels in poorer countries. From this, we may extend that higher income counties (in our situation) are more likely to follow a healthy lifestyle – not smoking, quality diet, regular exercise, etc.

In Length of Unemployment and Health-related Outcomes, Hammarstrom, Janlert, and Winefield studied the relationship between cumulative length of intermittent spells of unemployment and different health-related outcomes using data from a longitudinal study of school leavers in a mid-size town in northern Sweden. Respondents were followed for 14 years and the researchers concluded that 'cumulative length of unemployment is correlated with deteriorated health and health behavior'. Extended from this mid-size Swedish town to the US at a county-level, we may expect a strong, negative correlation between unemployment and healthy lifestyle.

Summary of Literature Review for Variables

The source, type, and name of each variable under consideration / for inclusion is captured below:

Table 1: Literature Review: Variable Summary

Source	Type	Variable
Bhardwaj, Amiri, Buchwald, and Amram	Dependent	Longevity
Kaminsky, Lessler, Sharfstein, and Wallace	Dependent	Obesity
Azevedo, Hallal, Wells, and Victoria	Dependent	Physical Activity
An, Li, and Jiang	Independent	EQI (Environmental Quality)
Holick and Wacker	Independent	Sunlight
Bremberg	Independent	Population Density
Ahern, Brown, and Dukas	Independent	Nutrition
Ferreira, Jardim, Jardim, Sousa, and Rosa	Independent	Alcohol Consumption
Islam, Leer, Teo, et al.	Independent	Income
Hammarstrom, Janlert, and Winefield	Independent	Unemployment

Additional Research for Model Building

Julian Faraway's *Linear Models with R* and Simon Sheather's *A Modern Approach to Regression with R* proved invaluable as references for applied regression, feature selection, model building and model selection.

All Literature Review sources are cited in the Bibliography section.

Methodology

Our research was focused on attempting to answer the following questions:

1. What United States counties are most favorable for an active, healthy lifestyle?
2. What are the differentiating characteristics that make them so?
3. What might the best regression model be for modeling the relationship between our healthy lifestyle metric and these differentiating characteristics?

Our methodology for answering these questions was a three step process: Data Gathering & Preprocessing, Data Exploration & Preparation, and Multivariate Regression Analysis. The nature of each is described briefly below:

- **Data Gathering & Preprocessing**: the gathering of data, creation of our (dependent) health score variable, identification of our healthiest counties, bringing of data (from numerous sources) into a consistent form so that all of our disjointed sets could be merged into one "master" dataframe.
- **Data Exploration & Preparation**: the investigation of our data's characteristics, including type, range of values, presence of missing values, correlation to dependent and other independent variables, and distributions. The handling of NA's and outliers, normalization of variable ranges onto a 0-to-1 scale, creation of features, and optimization of our initial model based on data transformations.
- **Multivariate Regression Analysis**: the creation / further exploration of linear and regularization (ridge and lasso) regression models as applied to raw and transformed data.

We train-test split our data, maintained evaluation metrics for each model, and then compared and contrasted each of our models using our evaluation metrics.

Data Gathering & Preprocessing

Dependent Variable Creation

In order to create our dependent 'health score' variable, I first familiarized myself with the data at hand.

Life expectancy, obesity, and physical activity data were downloaded from the Institute for Health Metrics and Evaluation and converted to a csv-compatible form, where then a subset of columns were selected for our consideration for the creation of a dependent 'health score' variable.

- For **life expectancy data** we read in: `Male life expectancy, 2010 (years)`, `Female life expectancy, 2010 (years)`, `Difference in male life expectancy, 1985-2010 (years)`, and `Difference in female life expectancy, 1985-2010 (years)`.
- For **obesity data** we read in: `Male obesity prevalence, 2009 (%)`, `Female obesity prevalence, 2009 (%)`, `Difference in male obesity prevalence, 2001-2009 (percentage points)`, and `Difference in female obesity prevalence, 2001-2009 (percentage points)`.
- For **physical activity data** we read in: `Male sufficient physical activity prevalence, 2009 (%)`, `Female sufficient physical activity prevalence, 2009 (%)`, `Difference in male sufficient physical activity prevalence, 2001-2009 (percentage points)`, and `Difference in female sufficient physical activity prevalence, 2001-2009 (percentage points)`.

For sake of conciseness, the majority of exploratory details and plots for our dependent variable's creation have been remitted (although the code is available in the Appendix). Prior to moving on though, it is worth noting a few points that were observed prior to the normalization and congregation of our sub-variables:

- With regard to **longevity**: on average, males live to be ~75 years old while females live to be ~80 years old. Thus, females live ~5yrs more than males on average. On average, male life expectancy increased by ~4 years while female life expectancy increased by ~1.5 years. Thus, male life expectancy increased at a greater rate than female life expectancy from 1985-2010.
- With regard to **obesity**: on average, 38% of females were obese whereas 36% of males were. Thus, females have a *slightly* higher incidence of obesity than males. On average, the male obesity rate increased by ~7.2% while the rate of female obesity increased by ~6.7%. Thus, males got fatter at a greater rate than females from 2001 to 2009.

- With regard to **physical activity**: on average, 55% of males vs. 48.7% of females received sufficient physical activity in 2009. Thus, males reported a higher level of physical activity. On average, males had a 1.9% increase in physical activity from 2001 to 2009 whereas females reported a 4.7% increase over the same period. Thus, females increased their activity levels at a greater rate than males from 2001 to 2009.

Our dependent variable is built upon these sub-variables. The above notes provide context regarding the variable upon which our regression model is to be fastened.

For each over-arching variable (life expectancy, obesity, or physical activity), we read in the 4 corresponding variables listed above, normalize, compile the over-arching variable as a summation of the normalized sub-variables and then normalize the result.

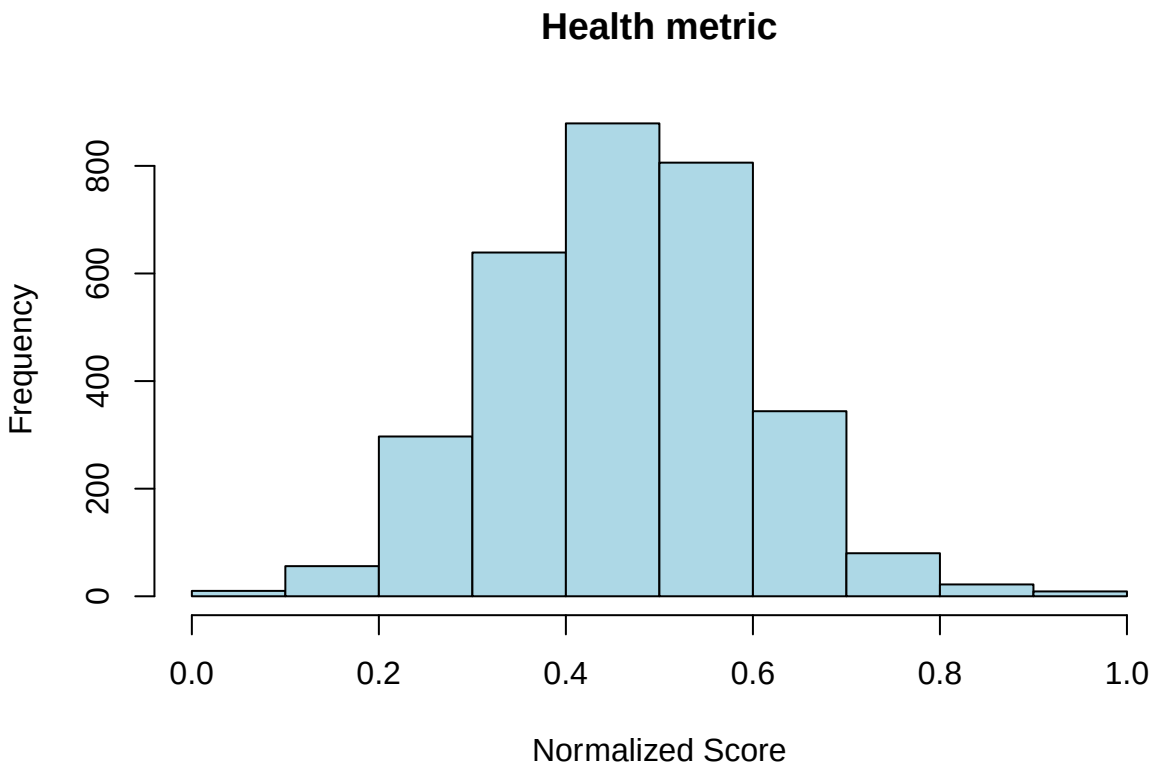
The normalization of sub-variables and (later) over-arching variables was done in order to bring all data to a 0-to-1 scale via the following equation:

$$Transformed.Values = \frac{(Values - Min)}{(Max - Min)}$$

Upon normalization of our over-arching variables, we created our dependent 'healthy lifestyle' variable as a combination of longevity, obesity, and physical activity:

$$Lifestyle = Normalized.Life - Normalized.Obesity + Normalized.Activity$$

The result was normalized, bringing our cycle of thrice normalizing and twice congregating to a close, with a dependent variable on a 0 to 1 scale:



```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.0000  0.3708  0.4642  0.4639  0.5546  1.0000
```

For our 'healthy lifestyle' metric, we observe a normal distribution whose peak is centered between 0.4 and 0.5. When we consult the summary statistics, we verify a mean of 0.4639 and a median of 0.4642. Confirming that our dependent variable is slightly left skewed.

Top 10 Healthiest Counties

As a next step, we utilize our health score metric to filter through county data for the top 10 healthiest counties:

State	County	health_score
Colorado	Routt	1.0000000
California	Marin	0.9889641
Colorado	Pitkin	0.9833580
Wyoming	Teton	0.9778477
Colorado	Eagle	0.9745991
California	San Francisco	0.9139634
Colorado	Summit	0.9072650
Colorado	Douglas	0.9054401
Utah	Summit	0.9014540
Colorado	Gunnison	0.8978052

From the above list, we note (6) Colorado, (2) California, (1) Utah and (1) Wyoming county. From this, we extend a few assumptions regarding factors that might come into play for the healthiest counties:

- sunshine,
- median income,
- sparser population clusters (aside from San Fransisco), and
- friendliness to an active, healthy lifestyle.

We note these factors with interest and plan to revisit our assumptions later. We can observe the variables that are included in our optimal multi regression model as well as those that appear to carry the most predictive impact (ie. largest coefficients).

With our health metric created and a brief investigation into the top 10 healthiest counties, we move on to reading in, exploring, and preparing our independent variables.

Independent Variable Preprocessing

We sought county-level data where the counties of the United States would form the basis of our observations and variables could include numerous standard and non-standard health-related metrics:

- `Alcohol Consumption` sourced from IHME.
- `Heart Disease` sourced from the IHME
- `Education`, `Unemployment`, and `Population` sourced from the USDA.
- `Environmental Quality Index` sourced from the EPA.
- `Food Insecurity` sourced **by request** from Feeding America.
- `Sunlight` sourced from CDC Wonder.
- `Poverty` and `MedianIncome` sourced from the US Census Bureau.

While the IHME is an independent global health research center associated with the University of Washington and Feeding America is one of the largest nonprofit organizations in the United States, every other data source is connected to the United States government. **The reputability and dependability of the source / institution motivated the election of these sources.**

Datasets were downloaded from their respective platforms, simplified to a .csv-compatible form (obviously impertinent variables and header rows were dropped), uploaded to Github and then read in via R's built-in read_csv() function. This operation was completed (9) times because the majority of our variables were listed in separate sets.

Bringing each of our sources to a consistent format was of the utmost importance for us to merge dataframes and (later) explore our data.

In order to do so:

- impertinent variables and observations were dropped,
- where `State` names were abbreviated they were converted to the full name,
- excess verbage was dropped from our `County` variable,
- percent change variables were added (where applicable),
- excess characters were dropped from observations (ie. "%" from percent change variables),
- variables were converted to the proper type (ie. numeric variable's listed as character type),
- the general consistency of variable names, type, and format was established, and
- all dataframes were merged based on matching `State` and `County` observations.

The result of our merging all dataframes / variables into one master is a 3154 observation x 25 variable dataframe `df`. We'll explore the resulting data frame in the next section to see what insight we might glean to better inform our data preparation and model-building.

Data Exploration & Preparation

The goal of exploratory data analysis (or EDA for short) is to really grasp and understand the data at hand.

For our approach, we got to know our data's structure and value ranges, visualized the relationship our variables had with one another and the target variable via correlation matrix, and explored the pertinence of each variable and the corresponding distribution for features we would select. We then prepared our data (handled NA's and outliers, normalized, and created features) and proceeded to optimize a linear regression model based on our transformed data.

High Level Exploration

We utilized the built-in glimpse() and summary() methods to gain insight into the dimensions, variable characteristics, and value ranges for our training dataset:

```
## Rows: 3,154
## Columns: 25
## $ State           <chr> "Alabama", "Alabama", "Alabama", "Alabama", "Alabama",~
## $ County          <chr> "Autauga", "Baldwin", "Barbour", "Bibb", "Blount", "Bu~
## $ health_score    <dbl> 0.4597290, 0.5185262, 0.2292573, 0.2954267, 0.2025867,~
## $ Hvy             <dbl> 14.7, 16.6, 16.9, 15.4, 14.3, 18.8, 16.8, 13.9, 17.7, ~
## $ HvyPctChg       <dbl> -2.7, 1.2, 17.5, 1.4, 3.1, 21.4, 8.7, 12.4, 10.9, 12.1~
## $ Bng             <dbl> 31.1, 30.4, 32.9, 31.9, 29.4, 38.1, 33.6, 31.0, 33.9, ~
```

```
## $ BngPctChg       <dbl> -1.1, -1.8, 3.4, -2.3, -3.4, 15.9, 8.1, 3.9, 0.0, -5.0~
## $ Mortality       <dbl> 316.36, 272.04, 255.09, 378.09, 307.90, 322.56, 382.55~
## $ MortalityChg    <dbl> -52.47, -36.95, -45.95, -31.16, -39.89, -68.95, -39.47~
## $ LTHighSchool    <dbl> 11.5, 9.2, 26.8, 20.9, 19.5, 25.3, 15.0, 15.6, 18.4, 1~
## $ HighSchool      <dbl> 33.6, 27.7, 35.6, 44.9, 33.4, 40.3, 45.2, 32.8, 36.7, ~
## $ SomeCollege     <dbl> 28.4, 31.3, 26.0, 23.8, 34.0, 22.3, 23.7, 33.2, 31.6, ~
## $ College         <dbl> 26.6, 31.9, 11.6, 10.4, 13.1, 12.1, 16.1, 18.5, 13.3, ~
## $ EQI             <dbl> 0.284681678, 0.754969418, -0.005637936, 0.304039747, 1~
## $ FoodInsecurity  <dbl> 15.6, 12.9, 21.9, 15.1, 13.6, 20.5, 19.1, 17.4, 16.4, ~
## $ Sun             <dbl> 18557.98, 19101.34, 18642.40, 18282.04, 17606.36, 1866~
## $ Unemployment    <dbl> 2.7, 2.7, 3.8, 3.1, 2.7, 3.6, 3.6, 3.5, 2.9, 2.9, 2.7,~
## $ UnemploymentChg <dbl> -2.4, -2.6, -4.5, -3.3, -2.7, -3.2, -3.3, -3.0, -2.5, ~
## $ Poverty         <dbl> 12.1, 10.1, 27.1, 20.3, 16.3, 30.0, 21.6, 17.2, 19.6, ~
## $ Income          <dbl> 58233, 59871, 35972, 47918, 52902, 31906, 39944, 47747~
## $ Population      <dbl> 55869, 223234, 24686, 22394, 57826, 10101, 19448, 1136~
## $ Births          <dbl> 624, 2304, 256, 240, 651, 109, 213, 1269, 354, 222, 55~
## $ Deaths          <dbl> 541, 2326, 312, 252, 657, 109, 272, 1532, 441, 343, 50~
## $ NetMig          <dbl> 254, 5377, -128, 41, 65, -73, -123, -461, -259, 303, 2~
## $ PopChg          <dbl> 1298, 40969, -2771, -521, 504, -813, -1499, -4967, -96~
```

We noted 3154 observations x 25 variables: 2 categorical variables, 23 numeric variables, significant NA counts for numerous variables, and quite a difference in the magnitude of values based on the variables.

Our **categorical variables** were `State`, our state identifier and `County`, our county identifier. These variables were not of much use for anything beyond identification and would be excluded.

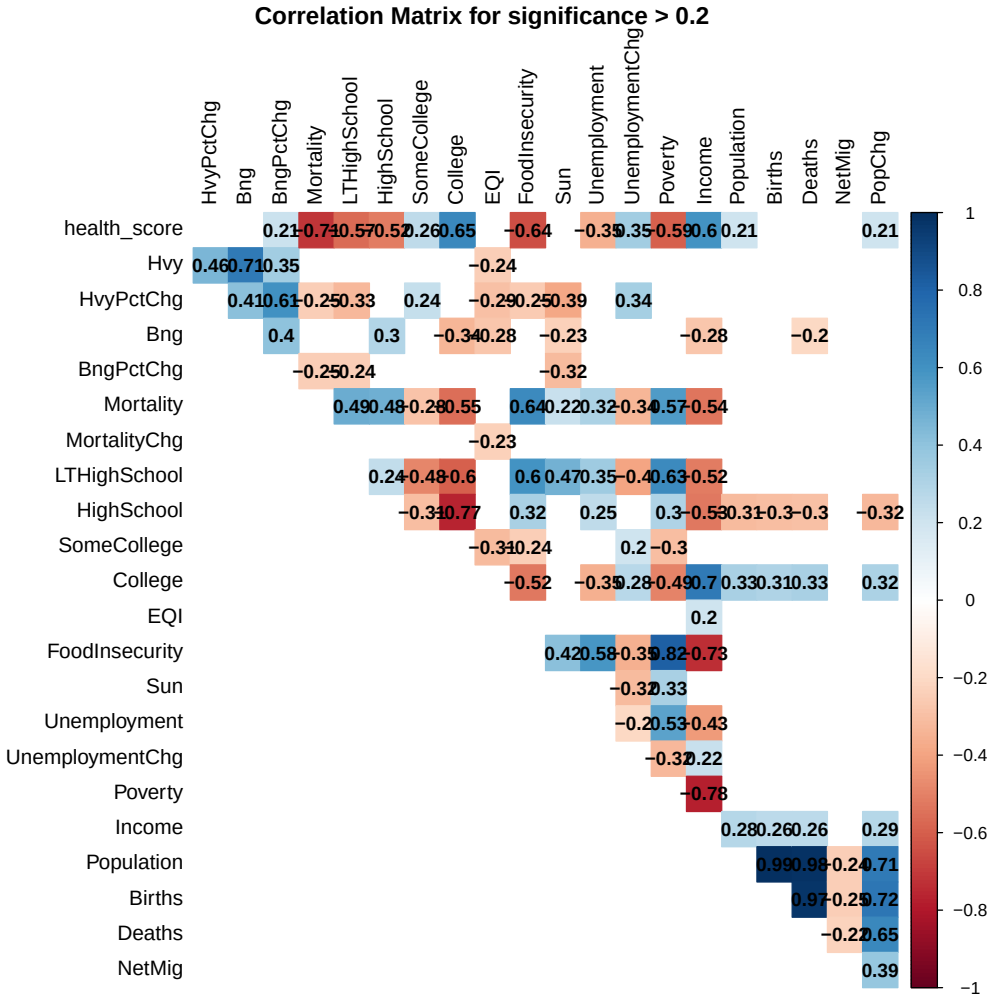
Our **dependent variable** `health_score` was quantitative and provided a measure of 'healthy lifestyle' as a function of longevity, obesity, and physical activity. This would be our "target" variable, the one against which we'd measure the impact of our independent variables.

As for our **quantitative, independent variables**:

- `Hvy`: % of population that drink heavily. *"Heavy" drinking is defined as the consumption, on average, of more than one drink per day for women or two drinks per day for men in the past 30 days.*
- `HvyPctChg`: % change in heavy drinking from 2005 to 2009.
- `Bng`: % of population that binge drink. *"Binge" drinking is defined as the consumption of more than four drinks for women or five drinks for men on a single occasion at least once in the past 30 days.*
- `BngPctChg`: % change in binge drinking from 2005 to 2009.
- `Mortality`: mortality rate in 2014 as a result of heart disease.
- `MortalityChg`: % change in mortality rate from 2005 to 2014 as a result of heart disease.
- `LTHighSchool`: % of population educated less than high school from 2015-2019.
- `HighSchool`: % of population educated at a high school level (and no further) from 2015-2019.
- `SomeCollege`: % of population who received some college education from 2015-2019.
- `College`: % of population who completed at least a Bachelor's level education from 2015-2019.
- `EQI`: Environmental Quality Index.
- `FoodInsecurity`: % of population whose food intake was disrupted by lack of resources in 2018.
- `Sun`: average daily sunlight (KJ/m^2) in 2011.
- `Unemployment`: rate of unemployment in 2019.
- `UnemploymentChg`: change in unemployment rate from 2015 to 2019.
- `Poverty`: rate of poverty in 2019.
- `Income`: median household income in 2019.
- `Population`: county-level population for 2019.
- `Births`: county-level births for 2019.
- `Deaths`: county-level deaths for 2019.
- `NetMig`: county-level net migration for 2019.
- `PopChg`: county-level rate of population change from 2010 to 2019.

Correlation

We dropped NA values from consideration (150 observations) and turned our attention to exploring the relationship these variables had with one another and with the target via correlation matrix. We considered only variables with a correlation significance > 0.2 in our plot:



From the above plot, we extended that multicollinearity was a concern. As such, we considered the elimination of a large portion of the variables we'd carried upto this point. To then proceed with only those with strong predictive promise and / or unique value added (ie. `BngPctChg` or `Sun`).

Based on these guidelines, it appeared that we should proceed with at least `BngPctChg`, `Mortality`, `College`, `FoodInsecurity`, `Unemployment`, `Income`, and `PopChg`. These variables were relatively unique from one another and had a strong correlation with `health_score`.

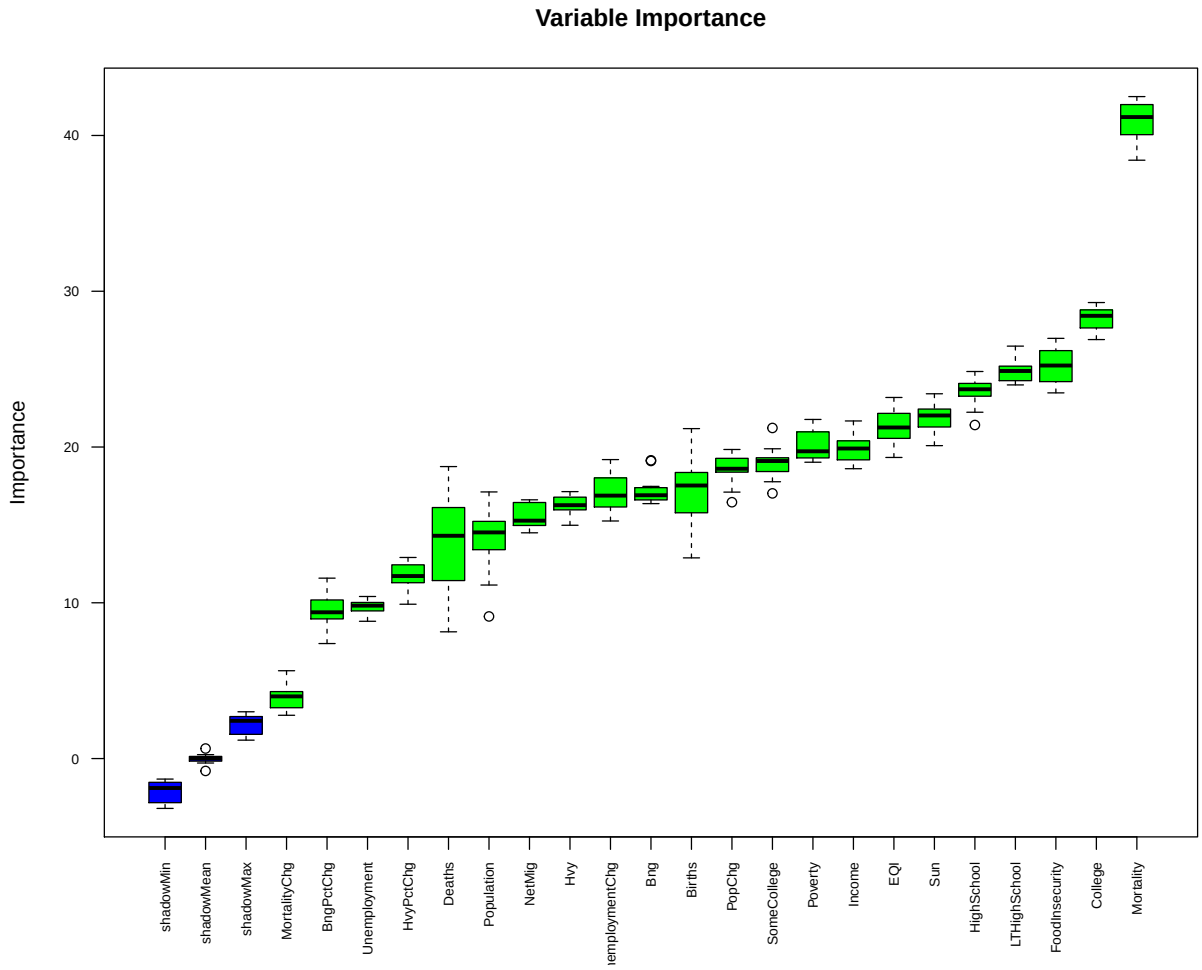
To clarify we consulted a table (code available in Appendix) with the proportion of missing data and correlation with our target variable. We found that none of our independent variables were missing data, that `Hvy`, `MortalityChg`, `NetMig` had a weak correlation with `health_score`, and that the remainder of our variables carried a relatively strong correlation with the target.

Feature Selection (Differentiating Characteristics)

To address weak target correlation and multicollinearity we considered the exclusion of 13 variables : `Hvy`, `MortalityChg`, and `NetMig` due to weak correlation with the target variable and `HvyPctChg` `Bng`,

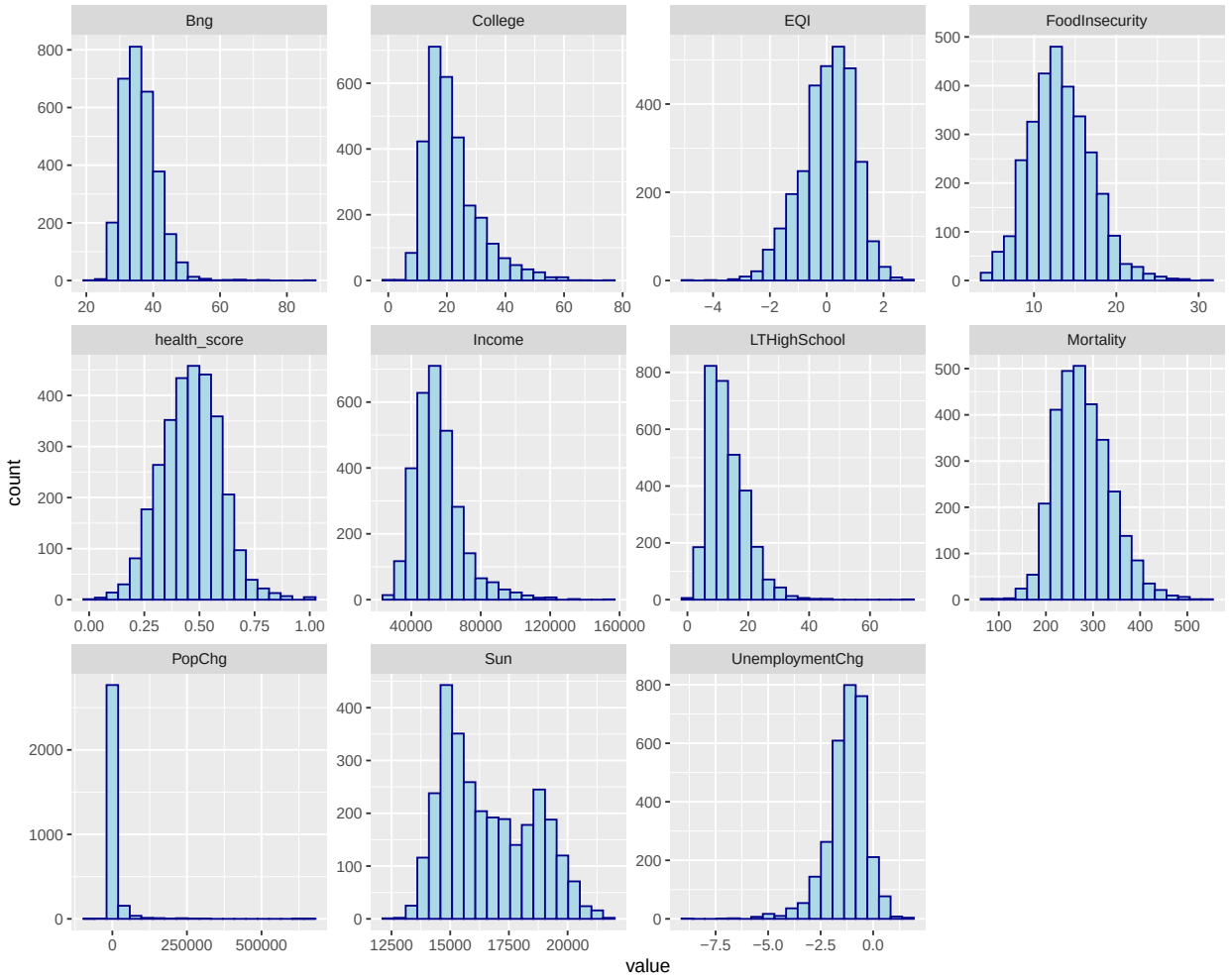
`LTHighSchool`, `HighSchool`, `SomeCollege`, `Unemployment`, `Poverty`, `Population`, `Births`, and `Deaths` due to multicollinearity.

Before doing so, we utilized the **Boruta** function for feature ranking and selection as confirmation. In doing so, `MortalityChg`, `BngPctChg`, `Unemployment`, `HvyPctChg`, and `Population` were identified as the variables with the lowest importance.



Additionally, we observed that `Mortality`, `College`, `FoodInsecurity`, `LTHighSchool`, `Sun`, `EQI`, `Income`, `PopChg`, `Bng`, and `UnemploymentChg` appeared to be our strongest predictors (with the lowest incidence of multicollinearity).

Based on variable importance, correlation with `health_score` and multicollinearity, we proceeded with these variables and visited their corresponding histograms:



From the histograms above, we noted a number of **non-normal** (ie. `PopChg`), **bimodal** (ie. `Sun`), **normal right skewed** (ie. `College`), **normal left skewed** (ie. `EQI`), and relatively **normal** (ie. `Mortality`) distributions.

The wide-ranging difference in scales and non-normal distributions could have proven problematic in applying generalized linear regression models, so we elected, as a part of our data preparation, to explore the impact of normalization. Additionally, we would explore the impact of outlier removal and feature creation (ie. converting non-normal variables to dummy "flag" variables) to address the issue of non-normal distributions.

Data Preparation

First, we generated a baseline model with our 10 independent variables and observed an **R^2 of 0.6603**. Once we'd established our baseline model and performance statistics, we set out to optimize the results.

We proceeded to use the R^2, the coefficient of determination, to assess the impact of each step on the strength of our model. If a step was seen to have no (or a negative) impact, it was deemed inessential. Whereas if a step was seen to have a positive impact, it was deemed essential. The results of each step are summarized below:

- **Handling NA's (Imputation)**: being that there were 0 missing values, this step was deemed inessential.
- **Normalization**: even though the normalization of variable scales had no positive impact on R^2, we maintained it for the sake of simpler model interpretations.

- **Outlier Handling**: being that the use of Cook's distance for outlier identification and handling had no positive impact on R^2 value (no matter how we varied our filter value), this step was deemed inessential.
- **Feature Creation**: being that the creation and inclusion of 7 additional features led to an (improved) R^2 of 0.6735, this step was deemed essential.

As a final transformation step, we utilized the stepAIC() function to optimize our model based on AIC score - a measure of goodness of fit *and* model complexity. During feature creation, we'd added 7 features and we wanted to ensure that each of these features was indeed improving our predictive capability rather than just making our model more complex.

After AIC optimization, model complexity was reduced from 17 to 16 independent variables, and predictive capability was maintained. It was a useful step that appears to have enhanced our model.

As a next step, we'd proceed through model building on to model selection.

Multivariate Regression Analysis

In order to strike the greatest balance between model complexity, goodness of fit and error, we explored two more linear regression models before moving on to regularization.

We explored linear regression models with raw data and AIC-optimized data, and then proceeded to use regularization (ridge and lasso) regression models on both our transformed and raw data. The aim was to explore different models and select the strongest amongst them. To do so, we train-test split our data (raw and transformed), constructed our models, and then evaluated each model's performance on seen and unseen data.

To conclude, we compared and contrasted the performance of all of our models using the R-squared and Root Mean Square Error (RMSE) as a part of our selection criteria.

Linear Regression (Raw)

As a means of comparing our (earlier) transformed model to a true baseline, a model with all of the original features and no operations performed, we considered a model with all 22 independent variables (from earlier).

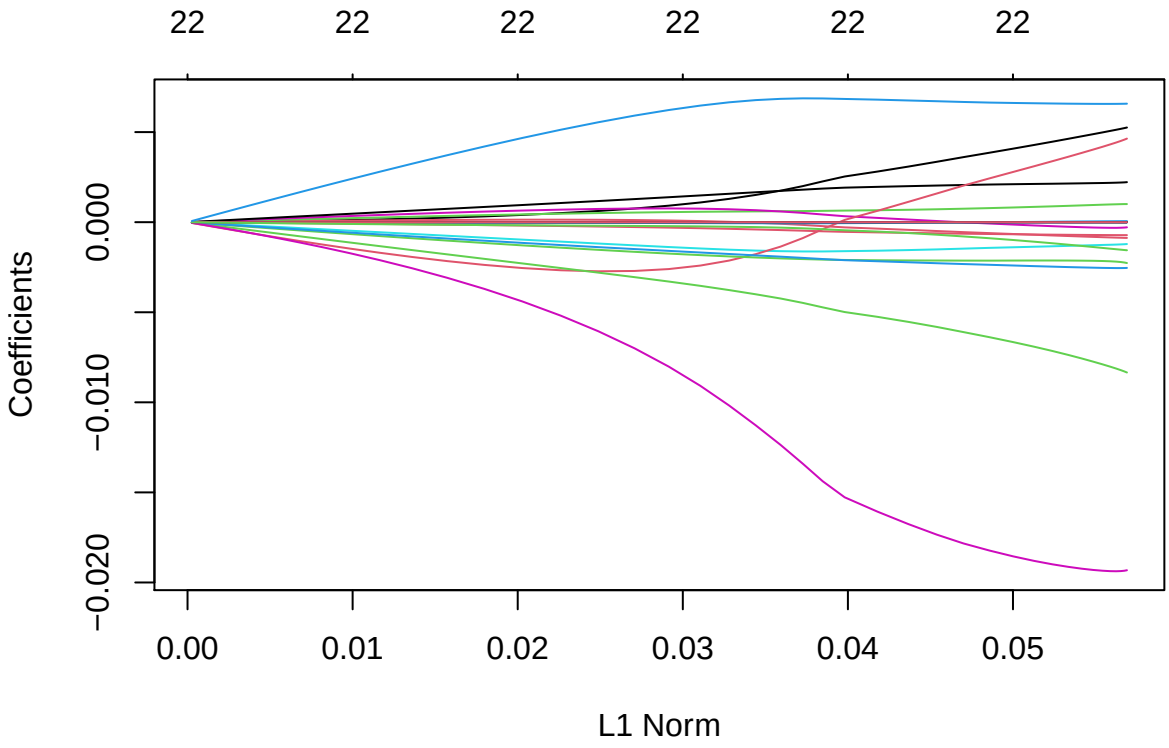
The perceived strength of this "raw" model would have been the number of features whereas the weakness would be the higher incidence of over-fitting.

With this in mind, we also considered an AIC-optimized model. A model where variables were *automatically* selected in a step-wise manner based on the predictive potential that their inclusion might bring.

Ridge Regression

Ridge regression is an extension of linear regression where model complexity and the potential for over-fitting, are addressed by adding a penalty parameter to penalize large coefficients.

One of the major differences between linear and regularized regression is the use of a `lambda` tuning parameter. To automate the task of finding the optimal lambda value, we made use of the cv.glmnet() function.



When lambda is zero, the penalty term has no effect, whereas when lambda increases to infinity, the shrinkage penalty grows and our coefficients approach zero. Our optimal lambda was computed as 0.003162278 (for raw data) and we can see from the above plot the critical role optimal lambda computation can have when evaluating our model.

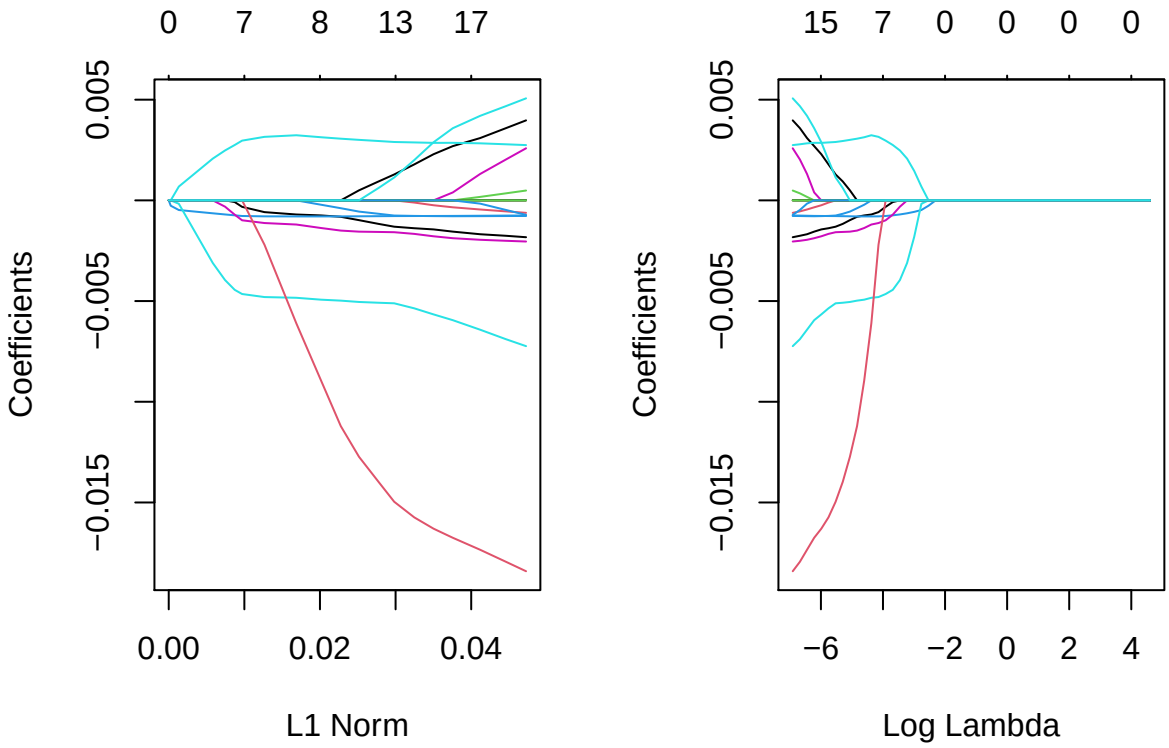
We used the glmnet package to build our regularized regression models (with `alpha` = 0 for ridge regression and `alpha` = 1 for lasso regression). Being that the corresponding function does not work with dataframes, we used the dummyVars() function from the caret package to create our model matrices and then used the predict() function to create numeric model matrices for both training and test data.

Our first ridge regression model was trained on raw data while the second was trained on transformed data. *Performance metrics are noted in the 'Model Selection' section.*

Lasso Regression

LASSO (Least Absolute Shrinkage and Selection Operator) regression is an extension of linear regression where model complexity and the potential for over-fitting, are addressed by limiting the sum of the absolute values of model coefficients.

Similar to ridge regression, the first step was to find the optimal lambda value and we automated this task through the use of the cv.glmnet() function.



When lambda approaches zero, the loss function of our model approaches the OLS function and we consider more variables, whereas when lambda grows, the regularization term has a greater effect and we see fewer variables in the model. Our optimal lambda was computed as 0.001 for raw data. We can observe in the plots above the effect varying lambda (on different scales) has on coefficient values.

A similar model-fitting approach to ridge regression was taken (utilizing the glmnet package), with our first LASSO model trained on raw data and the second trained on transformed data. *Performance metrics are noted in the 'Model Selection' section.*

Model Selection

We output the performance metrics of all models in a table to select the model with the strongest predictive promise. We evaluated performance using the R-squared value and Root Mean Squared Error (RMSE). Lower RMSE and higher R-squared values are indicative of a stronger model.

To succinctly summarize our model optimization attempts and results, we captured the method used, the data the method was applied to, as well as the number of variables considered, and corresponding evaluation metrics for raw and transformed data:

From the above model statistics, we can extend that:

- `Var_Num`: Models 3, 5, and 7 are favorable for model complexity being that they have the lowest variable count (16). Our data transformations appear to have led to the optimal level of simplicity, followed closely by the AIC-optimized raw linear model.
- `R2_train`: Model 4 is the highest performing, followed by Models 5, 2, 7, and then 3. A higher R squared value for this column is indicative of stronger predictive performance for (seen) training data. Regularization and raw (broader) data appears to be correlated with stronger predictive outcomes on

Table 2: Regression Model Comparison

Model	Method	Data	Var_Num	R2_train	RMSE_train	R2_test	RMSE_test
1	Linear	raw	22	0.6678	0.0759	0.6678	0.0827
2	Linear	raw(AIC)	17	0.673	0.0759	0.673	0.0842
3	Linear	transformed	16	0.671	0.0764	0.71	0.0815
4	Ridge	raw	22	0.6734	0.0761	0.659	0.0789
5	Ridge	transformed	16	0.6732	0.0764	0.632	0.0814
6	Lasso	raw	22	0.6681	0.0767	0.6642	0.0783
7	Lasso	transformed	16	0.6724	0.0765	0.6354	0.0811

training data. With that said, over-training can lead to over-fitting and thus this metric is a secondary measure to performance on unseen data.

- `RMSE_train`: Models 1, 2, and then 4 have the lowest error for (seen) training data,
- `R2_test`: Model 3 is the highest performing (by a relatively significant margin), followed by Models 2, 1, 6 and then 4. A higher R squared value for this column is indicative of stronger predictive performance on (unseen) testing data. Our data transformations appear to have led to the strongest predictive model for unseen data.
- `RMSE_test`: Models 6, 4, 7, 5, and then 3 have the lowest error for (unseen) testing data. Regularization methods appear to reduce error.

To find the point of balance between model complexity, goodness of fit and error, we elected Model 3. Model 3, linear regression applied to transformed data, was the most promising of models due to its simplicity, relatively low error rate (all models performed well here), and strong performance on unseen data. This last metric, unseen data performance, was the deciding factor.

Conclusion & Next Steps

What have I learned?

We started out with the goal of answering the following 3 questions:

1. What United States counties are most favorable for an active, healthy lifestyle?
2. What are the differentiating characteristics that make them so?
3. What might the best regression model be for modeling the relationship between our healthy lifestyle metric and these differentiating characteristics?

Being that readily-available data that would have suited the scope of this project was difficult to find, the scope and aim of the project expanded to include the creation of a healthy lifestyle metric and the compilation of a set of independent variables to be read in in conjunction. In other words, a large portion of this project was dedicated toward the creation of the dataset we would consider for questions 2 and 3.

In the data gathering and pre-processing phase, we created our dependent health score variable as a combination of life expectancy, obesity, and physical activity data read in from the IHME website. Over-arching metrics (a combination of 4 sub-metrics) were summed and normalized to bring our variable onto a 0 to 1 scale. We also utilized our health score to filter for the top 10 healthiest counties. We then extended a few assumptions regarding the factors that might come into play for the healthiest counties and we took these factors into consideration when reading in our independent variables. Finally, we read in county-level standard and non-standard health-related variables (ie. Alcohol Consumption, Heart Disease, and Education)

from different sources. We then brought the data from each of these sources into a consistent format so that we could merge data frames and explore our data.

In the data exploration and preparation phase, we first interpreted our 3154 observations x 25 variables. We then dropped impertinent variables and NA values, identified our strongest and weakest predictors (strongest predictors included: Mortality, College, FoodInsecurity, LTHighSchool, Sun, EQI, Income, PopChg, Bng, and UnemploymentChg), and proceeded with only 10 of the original 24 independent variables (due to multicollinearity and weak target-correlation). Through normalization, feature creation, and AIC-optimization, we optimized on the coefficient of determination (R-squared). We found our optimal (transformed) linear regression model included 16 independent variables with a coefficient of determination of 0.6734.

In the multivariate regression analysis phase, we expanded upon this transformed linear regression model. We set out to explore linear regression models with raw data and AIC-optimized data, and then proceeded to use regularization (ridge and lasso) regression models on both our transformed and raw data. The aim was to explore different models and select the strongest amongst them. To do so, we train-test split our data (raw and transformed), constructed our models, and then evaluated each model's performance on seen and unseen data. We then output the performance metrics of all models in a table, where the R-squared and RMSE values were used as part of our selection criterion. The linear regression model applied to transformed data was found to be the strongest model, largely due to its stand-out performance on unseen (testing) data.

Note: 3154 is greater than the 3009 observations which we'd noted as the number of US counties. We consider this broader range of "territories" (ie. in Alaska) as well where the observations are consistent across our dataset.

Areas for Future Research

Over the course of the project, while researching, and especially during the final phase, I realized a number of ways in which this project could have been taken to a higher level and I'd like to share these potential refinements here.

They include, but are not limited to:

1. Read in ready-made data. I spent a large portion of time gathering and pre-processing data, and one simple alternative would have been to read in ready-made data and then spend a larger portion of time on model-building or more advanced techniques.
2. Gather different variables. Gathering different variables would allow us to create a completely different dependent variable and/or take our models in a completely different direction. I read in some standard (ie. Mortality) and some non-standard variables (ie. Alcohol Consumption). Reading in a majority of non-standard variables could have led to more interesting data trends, models, and conclusions.
3. Read in more data. One of the simplest ways to improve the efficacy of a model is to incorporate more data. Especially more pertinent data. When we include more data that we'd hypothesized as having a strong impact on our dependent variable, it improves the likelihood that our models will perform with a greater predictive accuracy.
4. Experiment with more advanced regularization techniques (ie. elastic nets), explore bagging and bootstrapping, or change model classes completely to explore the impact on predictive power (ie. linear regression compared to neural nets). Either of these routes may have led to a model with a stronger performance than our Model 3.
5. Create more features. Feature creation led to the most significant improvement in our model (during Data Preparation). If I would've researched this realm further and allocated more time, it likely would have resulted in an even stronger model.
6. Consider classification rather than regression. If I would have utilized a binary health_score (ie. val $> 0.50 : 1$) then I could have explored numerous classification techniques on the same set of data. In some ways it would have been a completely different project that may have led to completely different insights.
7. Cast predictions. Beyond model selection, I could have applied Model 3 to cast predictions on our (unseen) test data to then assess how accurately it actually predicted healthy lifestyle at a county level.

Bibliography

In completing this research project, reference was made to the following

Data

1. IHME. United States Alcohol Use Prevalence by County 2002-2012. Retrieved from: http://ghdx.healthdata.org/record/ihme-data/united-states-alcohol-use-prevalence-county-2002-2012
2. IHME. United States Cardiovascular Disease Mortality Rates by County 1980-2014. Retrieved from: http://ghdx.healthdata.org/record/ihme-data/united-states-cardiovascular-disease-mortality-rates-county-1980-2014
3. USDA. Educational attainment for the U.S., States, and counties, 1970-2019. Retrieved from: https://www.ers.usda.gov/data-products/county-level-data-sets/download-data/
4. USDA. Poverty estimates for the U.S., States, and counties, 2019. Retrieved from: https://www.ers.usda.gov/data-products/county-level-data-sets/download-data/
5. USDA. Population estimates for the U.S., States, and counties, 2010-19. Retrieved from: https://www.ers.usda.gov/data-products/county-level-data-sets/download-data/
6. EPA. EPA Data Commons: 2006_2010_EQI_2Jan2018_VC.csv. Retrieved from: https://edg.epa.gov/EPADataCommons/public/ORD/CPHEA/EQI_2006_2010/
7. Feeding America. Map the Meal Gap Data. Retrieved from: https://www.feedingamerica.org/research/map-the-meal-gap/by-county (by request)
8. CDC. North America Land Data Assimilation System (NLDAS) Daily Sunlight (KJ/m^2) (1979-2011) Request. Retrieved from: https://wonder.cdc.gov/controller/datarequest/D80;jsessionid=097497410B08E44FD89ECC2AB08F?stage=results&action=sort&direction=MEASURE_DESCEND&measure=D80.M1
9. US Census Bureau. SAIPE State and County Estimates. Retrieved from: https://www.census.gov/data/datasets/2019/demo/saipe/2019-state-and-county.html

Literature

1. Bhardwaj, Amiri, Buchwald, and Amram (2020): Environmental Correlates of Reaching a Centenarian Age: Analysis of 144,665 Deaths in Washington State for 2011−2015
2. Ahern, Brown, and Dukas (2011): A National Study of the Association Between Food Environments and County Level Health Outcomes
3. An, Li, and Jiang (2017): Geographical Variations in the Environmental Determinants of Physical Inactivity among U.S. Adults
4. Holick and Wacker (2013): Sunlight and Vitamin D
5. Ferreira, Jardim, Jardim, Sousa, and Rosa (2019): Smoking, alcohol consumption and mental health: Data from the Brazilian study of Cardiovascular Risks in Adolescents (ERICA)
6. Islam, Leer, Teo (2013): Prevalance of a Healthy Lifestyle Among Individuals with Cardiovascular Disease in High-, Middle-, and Low-Income Countries
7. Azevedo, Hallal, Wells, and Victoria (2006): Adolescent physical activity and health: a systematic review
8. Kaminsky, Lessler, Sharfstein, and Wallace (2019): Comparison of US County-Level Health Performance Rankings with County Cluster and National Rankings
9. Bremberg (2020): Rural-urban mortality inequalities in four Nordic welfare states
10. Hammarstrom, Janlert, and Winefield (2014): Length of unemployment and health-related outcomes

Guiding Resources

- Datanovia. (2020). How to Normalize and Standardize Data in R for Great Heatmap Visualization [article]. Retrieved from: https://www.datanovia.com/en/blog/how-to-normalize-and-standardize-data-in-r-for-great-heatmap-visualization/

- Deepika Singh. (2019). Linear, Lasso, and Ridge Regression with R [article]. Retrieved from: https://www.pluralsight.com/guides/linear-lasso-and-ridge-regression-with-r
- Faraway (2015, pp. 25-49): Linear Models with R Faraway [text]
- Sheather (2009, pp. 263-293): A Modern Approach to Regression with R [text]

Appendix with Code

Data gathering and pre-processing

Being that data gathering and pre-processing was a major focus of this project, I've dedicated a separate RPubs publication to document the creation of our dependent 'healthy lifestyle' metric as well as the pre-processing of our independent variables. *Note: some light adaptations to this original code have been made above.*

All code (no output)

```
#Import libraries
library(tidyverse)
library(dplyr)
library(readr)
library(ggplot2)
library(RCurl)
library(rvest)
library(stringr)
library(tidyr)
library(kableExtra)
library(BBmisc)
library(tm)
library(sqldf)
library(inspectdf)
library(corrplot)
library(MASS)
library(caret)
library(glmnet)

options(scipen = 9)
set.seed(123)

#---User-defined function(s)---#

#Adapted correlation matrix used in EDA section:
plot_corr_matrix <- function(dataframe, significance_threshold){
  title <- paste0('Correlation Matrix for significance > ',
                  significance_threshold)

  df_cor <- dataframe %>% mutate_if(is.character, as.factor)

  df_cor <- df_cor %>% mutate_if(is.factor, as.numeric)
  #run a correlation and drop the insignificant ones
```

```
  corr <- cor(df_cor, use = 'na.or.complete')
  #prepare to drop duplicates and correlations of 1
  corr[lower.tri(corr,diag=TRUE)] <- NA
  #drop perfect correlations
  corr[corr == 1] <- NA
  #turn into a 3-column table
  corr <- as.data.frame(as.table(corr))
  #remove the NA values from above
  corr <- na.omit(corr)
  #select significant values
  corr <- subset(corr, abs(Freq) > significance_threshold)
  #sort by highest correlation
  corr <- corr[order(-abs(corr$Freq)),]
  #print table
  # print(corr)
  #turn corr back into matrix in order to plot with corrplot
  mtx_corr <- reshape2::acast(corr, Var1~Var2, value.var="Freq")

  #plot correlations visually
  corrplot(mtx_corr,
           title=title,
           mar=c(0,0,1,0),
           method='color',
           tl.col="black",
           na.label= " ",
           addCoef.col = 'black',
           number.cex = .9)
}

#Evaluation metrics used in Model Building / Selection section:
eval_metrics = function(model, df, predictions, target){
    resids = df[,target] - predictions
    resids2 = resids**2
    N = length(predictions)
    r2 = as.character(round(summary(model)$r.squared, 4))
    adj_r2 = as.character(round(summary(model)$adj.r.squared, 4))
    print(adj_r2) #Adjusted R-squared
    print(as.character(round(sqrt(sum(resids2)/N), 4))) #RMSE
}

## --- Model Selection --- ##

#Create Kable table to succinctly summarize variable research
Source <- c('Bhardwaj, Amiri, Buchwald, and Amram', 'Kaminsky, Lessler, Sharfstein, and Wallace', 'Azev
Type <- c('Dependent', 'Dependent', 'Dependent', 'Independent', 'Independent', 'Independent', 'Independe
Variable <- c('Longevity', 'Obesity', 'Physical Activity', 'EQI (Environmental Quality)', 'Sunlight', '

LRVS <- cbind(Source, Type, Variable)

LRVS %>%
  kbl(caption = "Literature Review: Variable Summary") %>%
  kable_minimal() %>%
  kable_styling(latex_options = "hold_position")
```

```
## --- Dependent Variable Creation --- ##

#Read in life expectancy data, convert to tibble, and select pertinent columns:
longevity <- read_csv("https://raw.githubusercontent.com/Magnus-PS/DATA-698/data/IHME_LifeExpectancy.csv
life_table <- as_tibble(longevity)
life_table <- life_table %>% dplyr::select(1:2,13:16) %>% na.omit()

#Read in obesity data, convert to tibble, and select pertinent columns:
obesity <- read_csv("https://raw.githubusercontent.com/Magnus-PS/DATA-698/data/IHME_Obesity.csv")
obesity_table <- as_tibble(obesity)
obesity_table <- obesity_table %>% dplyr::select(1:2,5:6,9:10) %>% na.omit()

#Read in physical activity data, convert to tibble, and select pertinent columns:
activity <- read_csv("https://raw.githubusercontent.com/Magnus-PS/DATA-698/data/IHME_PhysicalActivity.cs
act_table <- as_tibble(activity)
act_table <- act_table %>% dplyr::select(1:2,5:6,9:10) %>% na.omit()

###LIFE EXPECTANCY DATA: exploration, normalization, and compilation

#Explore life expectancy data at a county level:
glimpse(life_table)
summary(life_table)

#Extract variables of interest
m1 <- life_table$`Male life expectancy, 2010 (years)`
f1 <- life_table$`Female life expectancy, 2010 (years)`
dm1 <- life_table$`Difference in male life expectancy, 1985-2010 (years)`
df1 <- life_table$`Difference in female life expectancy, 1985-2010 (years)`

#Normalize data scale to be from 0 to 1
n_m1 = (m1-min(m1))/(max(m1)-min(m1))
n_f1 = (f1-min(f1))/(max(f1)-min(f1))
n_dm1 = (dm1-min(dm1))/(max(dm1)-min(dm1))
n_df1 = (df1-min(df1))/(max(df1)-min(df1))

#Histogram of original vs. normalized data
##Life expectancy histograms
par(mfrow=c(2,2))
hist(m1, breaks=10, xlab="Age (years)", col="lightblue", main="Male life expectancy, 2010")
hist(n_m1, breaks=10, xlab="Normalized Age (years)", col="lightblue", main="Male life expectancy, 2010")
hist(f1, breaks=10, xlab="Age (years)", col="lightblue", main="Female life expectancy, 2010")
hist(n_f1, breaks=10, xlab="Normalized Age (years)", col="lightblue", main="Female life expectancy, 201

##Longevity improvement histograms
par(mfrow=c(2,2))
hist(dm1, breaks=10, xlab="Age (years)", col="lightblue", main="Male longevity improvement, 1985-2010")
hist(n_dm1, breaks=10, xlab="Normalized Age (years)", col="lightblue", main="Male longevity improvement
hist(df1, breaks=10, xlab="Age (years)", col="lightblue", main="Female longevity improvement, 1985-2010
hist(n_df1, breaks=10, xlab="Normalized Age (years)", col="lightblue", main="Female longevity improvemen

#Add normalized variables together
life <- n_m1 + n_dm1 + n_f1 + n_df1
```

```r
#Normalize activity to 0-1 range
n_life = (life-min(life))/(max(life)-min(life))
#head(n_life)

#Histogram of original vs. normalized data
#par(mfrow=c(1,2))
#hist(life, breaks=10, xlab="Score", col="lightblue", main="Longevity metric")
hist(n_life, breaks=10, xlab="Normalized Score", col="lightblue", main="Longevity metric")
summary(n_life) #slight left skew

###OBESITY DATA: exploration, normalization, and compilation

#Explore obesity data at a county level:
glimpse(obesity_table)
summary(obesity_table)

#Extract variables of interest
m2 <- obesity_table$`Male obesity prevalence, 2009 (%)`
f2 <- obesity_table$`Female obesity  prevalence, 2009 (%)`
dm2 <- obesity_table$`Difference in male obesity prevalence, 2001-2009 (percentage points)`
df2 <- obesity_table$`Difference in female obesity prevalence, 2001-2009 (percentage points)`

#Normalize
n_m2 = (m2-min(m2))/(max(m2)-min(m2))
n_f2 = (f2-min(f2))/(max(f2)-min(f2))
n_dm2 = (dm2-min(dm2))/(max(dm2)-min(dm2))
n_df2 = (df2-min(df2))/(max(df2)-min(df2))

#Histogram of original vs. normalized data
par(mfrow=c(2,2))
hist(m2, breaks=10, xlab="Obesity rate (%)", col="lightblue", main="Male obesity prevalence, 2009")
hist(n_m2, breaks=10, xlab="Normalized obesity rate (%)", col="lightblue", main="Male obesity prevalenc
hist(f2, breaks=10, xlab="Obesity rate (%)", col="lightblue", main="Female obesity prevalence, 2009")
hist(n_f2, breaks=10, xlab="Normalized obesity rate (%)", col="lightblue", main="Female obesity prevale

par(mfrow=c(2,2))
hist(dm2, breaks=10, xlab="Obesity rate (%)", col="lightblue", main="Male obesity increase, 2001-2009")
hist(n_dm2, breaks=10, xlab="Normalized obesity rate (%)", col="lightblue", main="Male obesity increase
hist(df2, breaks=10, xlab="Obesity rate (%)", col="lightblue", main="Female obesity increase, 2001-2009
hist(n_df2, breaks=10, xlab="Normalized obesity rate (%)", col="lightblue", main="Female obesity increas

#Add normalized variables together
fat <- n_m2 + n_dm2 + n_f2 + n_df2

#Normalize activity to 0-1 range
n_fat = (fat-min(fat))/(max(fat)-min(fat))

#head(n_fat)

# Histogram of original vs. normalized data
#par(mfrow=c(1,2))
#hist(fat, breaks=10, xlab="Score", col="lightblue", main="Obesity metric")
hist(n_fat, breaks=10, xlab="Normalized Score", col="lightblue", main="Obesity metric")
```

```
summary(n_fat) #right skewed

###PHYSICAL ACTIVITY DATA: exploration, normalization, and compilation

glimpse(act_table)
summary(act_table)

#Explore and normalize male physical activity data
m3 <- act_table$`Male sufficient physical activity  prevalence, 2009 (%)`
f3 <- act_table$`Female sufficient physical activity  prevalence, 2009 (%)`
dm3 <- act_table$`Difference in male sufficient physical activity prevalence, 2001-2009 (percentage poi
df3 <- act_table$`Difference in female sufficient physical activity prevalence, 2001-2009 (percentage p

#Normalized Data
n_m3 = (m3-min(m3))/(max(m3)-min(m3))
n_f3 = (f3-min(f3))/(max(f3)-min(f3))
n_dm3 = (dm3-min(dm3))/(max(dm3)-min(dm3))
n_df3 = (df3-min(df3))/(max(df3)-min(df3))

#Histogram of original vs. normalized data
par(mfrow=c(2,2))
hist(m3, breaks=10, xlab="Physical activity rate (%)", col="lightblue", main="Male activity prevalence,
hist(n_m3, breaks=10, xlab="Normalized physical activity rate (%)", col="lightblue", main="Male activit
hist(f3, breaks=10, xlab="Physical activity rate (%)", col="lightblue", main="Female activity prevalenc
hist(n_f3, breaks=10, xlab="Normalized obesity rate (%)", col="lightblue", main="Female activity preval

par(mfrow=c(2,2))
hist(dm3, breaks=10, xlab="Physical activity rate (%)", col="lightblue", main="Male activity difference
hist(n_dm3, breaks=10, xlab="Normalized physical activity rate (%)", col="lightblue", main="Male activi
hist(df3, breaks=10, xlab="Physical activity rate (%)", col="lightblue", main="Female activity differen
hist(n_df3, breaks=10, xlab="Normalized obesity rate (%)", col="lightblue", main="Female activity diffe

#Add normalized variables together
active <- n_m3 + n_dm3 + n_f3 + n_df3

#Normalize activity to 0-1 range
n_active = (active-min(active))/(max(active)-min(active))

#head(n_active)

# Histogram of original vs. normalized data
#par(mfrow=c(1,2))
#hist(active, breaks=10, xlab="Score", col="lightblue", main="Physical activity metric")
hist(n_active, breaks=10, xlab="Normalized Score", col="lightblue", main="Physical activity metric")

summary(n_active) #slight right skew


###DEPENDENT VARIABLE CREATION: sum, normalize, and plot

#Add normalized variables together
lifestyle <- n_life - n_fat + n_active
```

```r
#Normalize health to 0-1 range
normalized_lifestyle = (lifestyle-min(lifestyle))/(max(lifestyle)-min(lifestyle))

#Histogram of original vs. normalized data
#par(mfrow=c(1,2))
#hist(lifestyle, breaks=10, xlab="Score", col="lightblue", main="Health metric")
hist(normalized_lifestyle, breaks=10, xlab="Normalized Score", col="lightblue", main="Health metric")

summary(normalized_lifestyle)
#head(normalized_lifestyle)

#create new df with state | county | health score
starter_df <- life_table %>%
    dplyr::select(1:2)

starter_df$health_score <- normalized_lifestyle
healthiest_counties <- filter(starter_df, `health_score` > 0.895) #top 10
healthiest_counties <- healthiest_counties[order(-healthiest_counties$`health_score`),] #descending ord

#head(starter_df)
#nrow(healthiest_counties) #10
healthiest_counties %>%
  kbl() %>%
  kable_minimal() %>%
  kable_styling(latex_options = "hold_position")


## --- Independent Variable Preprocessing --- ##

#Read in alcohol data, convert to tibble, and drop impertinent observation:
alcohol <- read_csv("https://raw.githubusercontent.com/Magnus-PS/DATA-698/data/AlcoholConsumption.csv")
alcohol <- as_tibble(alcohol)
alcohol <- alcohol[-1,] #drop State = National
#dim(alcohol) #3178 x 6

alcohol <- alcohol[alcohol$Location != alcohol$State,] #drop state observations from Location column
#dim(alcohol) #3178 - 3127 = 51 observations dropped

#rename columns
alcohol <- alcohol %>% rename(
    County = Location,
    Hvy = Hvy_2012,
    Bng = Binge_2012,
    HvyPctChg = `HvyPctChg_2005-2012`,
    BngPctChg = `BingePctChg_2005-2012`)

#drop excess verbage from County column
stopwords <- c("and", "Area", "Borough", "Census", "City", "County", "Division", "Municipality", "Paris
alcohol$County <- removeWords(alcohol$County, stopwords)
#head(alcohol)

#Read in heart data, convert to tibble, and drop impertinent observation:
heart <- read_csv("https://raw.githubusercontent.com/Magnus-PS/DATA-698/data/CardiovascularDisease.csv"
```

```
heart <- as_tibble(heart)
heart <- subset(heart, select = -c(`Mortality Rate, 2010*`)) #drop 2010
heart <- heart[-1,] #drop State = National
#dim(heart) #3193 x 4

# remove State == Location
heart <- heart[heart$Location != heart$State,]
dim(heart) #3193 - 3143 = 50 observations removed

# retitle columns
heart <- heart %>% rename(
    County = Location,
    Mortality_2005 = `Mortality Rate, 2005*`,
    Mortality_2014 = `Mortality Rate, 2014*`)

# retain value EXCLUSIVELY for Mortality Rate columns
heart$Mortality_2005 <- gsub("\\s*\\([^\\)]+\\)","",as.character(heart$Mortality_2005))
heart$Mortality_2014 <- gsub("\\s*\\([^\\)]+\\)","",as.character(heart$Mortality_2014))

#convert columns to proper type
heart$Mortality_2005 <- as.double(heart$Mortality_2005)
heart$Mortality_2014 <- as.double(heart$Mortality_2014)

#drop excess verbage from County column
heart$County <- removeWords(heart$County, stopwords)
heart$County <- gsub("(.*),.*", "\\1", heart$County) #remove everything after comma

# add Chg column
heart$MortalityChg <- heart$Mortality_2014 - heart$Mortality_2005

#finalize format of df
heart <- subset(heart, select = -c(`Mortality_2005`)) #drop 2005
heart <- heart %>% rename( Mortality = Mortality_2014)

#head(heart)

#Read in education data, convert to tibble, and drop impertinent observation:
education <- read_csv("https://raw.githubusercontent.com/Magnus-PS/DATA-698/data/Education.csv")
education <- as_tibble(education)
education <- education[-1,] #drop State = National

education$State <- state.name[match(education$State, state.abb)] #convert state abbreviation to name

education <- education[education$`Area name` != education$State,] #drop state observations from Area na
#dim(education) #3281 - 3233 = 48 observations dropped

#rename columns
education <- education %>% rename(
    County = `Area name`,
    LTHighSchool = `Percent of adults with less than a high school diploma, 2015-19`,
    HighSchool = `Percent of adults with a high school diploma only, 2015-19`,
    SomeCollege = `Percent of adults completing some college or associate's degree, 2015-19`,
    College = `Percent of adults with a bachelor's degree or higher, 2015-19`)
```

```r
#drop excess verbage from County column
education$County <- removeWords(education$County, stopwords)

#head(education) #verify

#Read in eqi data and convert to tibble:
eqi <- read_csv("https://raw.githubusercontent.com/Magnus-PS/DATA-698/data/EnvironmentalQualityIndex.csv
eqi <- as_tibble(eqi)
eqi <- subset(eqi, select = -c(3:7)) #drop indices that makeup EQI score

eqi$State <- state.name[match(eqi$State, state.abb)] #convert state abbreviation to name

#rename columns
eqi <- eqi %>% rename(
    County = County_Name,
    EQI = environmental_quality_index)

#drop excess verbage from County column
eqi$County <- removeWords(eqi$County, stopwords)

#head(eqi) #verify
#dim(eqi) #3281 x 6

#Read in food insecurity data and convert to tibble:
food <- read_csv("https://raw.githubusercontent.com/Magnus-PS/DATA-698/data/FoodInsecurity.csv")
food <- as_tibble(food)
#head(food)

#drop FIPS
food <- subset(food, select=-c(FIPS))
#dim(food) #3142 x 3: no need to drop observations

#convert State to full name
food$State <- state.name[match(food$State, state.abb)] #convert state abbreviation to name

#rename columns
food <- food %>% rename(
    County = `County, State`,
    FoodInsecurity = `2018 Food Insecurity Rate`)

#remove excess verbage from County
food$County <- removeWords(food$County, stopwords)
food$County <- gsub("(.*),.*", "\\1", food$County) #remove everything after comma

#drop % from Food Insecurity and convert to double
food$FoodInsecurity = as.double(gsub("[\\%,]", "", food$FoodInsecurity))

#head(food) #verify

#Read in sun data and convert to tibble:
sun <- read_csv("https://raw.githubusercontent.com/Magnus-PS/DATA-698/data/Sunlight.csv")
sun <- as_tibble(sun)
#dim(sun) #3161 x 3
```

```r
#rename column
sun <- sun %>% rename(Sun = `Avg Daily Sunlight`)

#drop excess verbage from County column
sun$County <- removeWords(sun$County, stopwords)
sun$County <- gsub("(.*),.*", "\\1", sun$County) #remove everything after comma

#head(sun) #verify

#Read in une ployment data and convert to tibble:
unemp <- read_csv("https://raw.githubusercontent.com/Magnus-PS/DATA-698/data/Unemployment.csv")
unemp <- as_tibble(unemp)
unemp <- unemp[-1,] #drop State = National
unemp <- subset(unemp, select=-c(3)) #drop 2016

unemp$State <- state.name[match(unemp$State, state.abb)] #convert state abbreviation to name
unemp <- unemp[unemp$area_name != unemp$State,] #drop state observations from Area name column

unemp <- unemp %>% rename(
    County = area_name,
    Unemployment = Unemployment_rate_2019,
    UnemploymentChg = `Unemployment_chg_2016-2019`) #rename columns

unemp$County <- removeWords(unemp$County, stopwords) #drop excess verbage from County column
unemp$County <- gsub("(.*),.*", "\\1", unemp$County) #remove everything after comma

#dim(unemp) #3274 - 3224 = 50 dropped
#head(unemp) #verify

#Read in wealth data and convert to tibble:
wealth <- read_csv("https://raw.githubusercontent.com/Magnus-PS/DATA-698/data/Wealth.csv")
wealth <- as_tibble(wealth)
wealth <- wealth[-1,] #drop State = National
#dim(wealth) #3193 x 4

wealth$State <- state.name[match(wealth$State, state.abb)] #convert state abbreviation to name
wealth <- wealth[wealth$County != wealth$State,] #drop state observations from Area name column
#dim(wealth) #3193 - 3143 = 50 observations dropped

#convert columns to proper type
wealth$PovertyRate <- as.double(wealth$PovertyRate)
wealth$MedianHouseholdIncome <- as.numeric(gsub(",","",wealth$MedianHouseholdIncome))

#rename columns
wealth <- wealth %>% rename(
    Poverty = PovertyRate,
    Income = MedianHouseholdIncome) #rename columns

wealth$County <- removeWords(wealth$County, stopwords) #drop excess verbage from County column

#head(wealth)

#Read in population data and convert to tibble:
```

```
pop <- read_csv("https://raw.githubusercontent.com/Magnus-PS/DATA-698/data/population.csv")
pop <- as_tibble(pop)

# rename columns
pop <- pop %>% rename(
    State = STNAME,
    County = CTYNAME,
    Pop_2010 = CENSUS2010POP,
    Population = POPESTIMATE2019,
    Births = BIRTHS2019,
    Deaths = DEATHS2019,
    NetMig = NETMIG2019) #rename columns

#add population change variable
pop$PopChg <- pop$Population - pop$Pop_2010

pop <- subset(pop, select=-c(3)) #drop 2010

#dim(pop) #3193 x 7
pop <- pop[pop$County != pop$State,] #drop state observations from Area name column
#dim(pop) #3193 - 3141 = 52 observations dropped

pop$County[1802] <- "Dona Ana County" #invalid UTF-8
pop$County <- removeWords(pop$County, stopwords) #drop excess verbage from County column

#head(pop)

## --- Merge df's --- ##

##1. merge health score and alcohol df's
#Trim white space
alcohol$County <- trimws(alcohol$County)
starter_df$County <- trimws(starter_df$County)
#SQL join
df <- sqldf("SELECT *
             FROM starter_df
             LEFT JOIN alcohol ON starter_df.State = alcohol.State AND starter_df.County = alcohol.Coun
#Remove extra State, County columns
df <- subset(df, select=-c(4,5))

##2. merge heart to df
#Trim white space
heart$County <- trimws(heart$County)
#SQL join
df <- sqldf("SELECT *
             FROM df
             LEFT JOIN heart ON df.State = heart.State AND df.County = heart.County")
#remove extra State, County columns
df <- subset(df, select=-c(8,9))

##3. merge education to df
#Trim white space
education$County <- trimws(education$County)
```

```
#SQL join
df <- sqldf("SELECT *
             FROM df
             LEFT JOIN education ON df.State = education.State AND df.County = education.County")
#remove extra State, County columns
df <- subset(df, select=-c(10,11))

##4. merge eqi to df
#Trim white space
eqi$County <- trimws(eqi$County)
#SQL join
df <- sqldf("SELECT *
             FROM df
             LEFT JOIN eqi ON df.State = eqi.State AND df.County = eqi.County")
#remove extra State, County columns
df <- subset(df, select=-c(14,15))

##5. merge food to df
#Trim white space
food$County <- trimws(food$County)
#SQL join
df <- sqldf("SELECT *
             FROM df
             LEFT JOIN food ON df.State = food.State AND df.County = food.County")
#remove extra State, County columns
df <- subset(df, select=-c(15,16))

##6. merge sun to df
#Trim white space
sun$County <- trimws(sun$County)
#SQL join
df <- sqldf("SELECT *
             FROM df
             LEFT JOIN sun ON df.State = sun.State AND df.County = sun.County")
#remove extra State, County columns
df <- subset(df, select=-c(16,17))

##7. merge unemp to df
#Trim white space
unemp$County <- trimws(unemp$County)
#SQL join
df <- sqldf("SELECT *
             FROM df
             LEFT JOIN unemp ON df.State = unemp.State AND df.County = unemp.County")
#remove extra State, County columns
df <- subset(df, select=-c(17,18))

##8. merge wealth to df
#Trim white space
wealth$County <- trimws(wealth$County)
#SQL join
df <- sqldf("SELECT *
             FROM df
```

```
            LEFT JOIN wealth ON df.State = wealth.State AND df.County = wealth.County")
#remove extra State, County columns
df <- subset(df, select=-c(19,20))

##9. merge pop to df
#Trim white space
pop$County <- trimws(pop$County)
#SQL join
df <- sqldf("SELECT *
            FROM df
            LEFT JOIN pop ON df.State = pop.State AND df.County = pop.County")
#remove extra State, County columns
df <- subset(df, select=-c(21,22))

##verify variables and dimensions
head(df)
dim(df) #3154 x 25

## --- EDA --- ##

#baseline EDA: glimpse() and summary()
glimpse(df)
#summary(df)


#drop NAs from consideration
df <- drop_na(df)
#dim(df) #3154 - 3004 = 150 dropped observations


#Select numeric variables
df_num <- as.data.frame(df[3:25])
#dim(df_num)
#head(df_num)

#Utilize custom-built correlation matrix generation function
plot_corr_matrix(df_num, 0.2)


#Compute proportion of missing data per variable
v <- colnames(df_num)
incomplete <- function(x) sum(!complete.cases(x)) / 3004
Missing_Data <- sapply(df_num[v], incomplete)
#head(Missing_Data) #verify

#Compute correlation between each variable and TARGET
target_corr <- function(x, y) cor(y, x, use = "na.or.complete")
HealthScore_Corr <- sapply(df_num[v], target_corr, y=df_num$health_score)
#head(HealthScore_Corr) #verify

#Bind and output Missing Data and Correlation with Target
MDHSC <- data.frame(cbind(Missing_Data, HealthScore_Corr))
MDHSC %>%
```

```
  kbl(caption = "Proportion of Missing Data vs. Correlation with Health Score") %>%
  kable_minimal() %>%
  kable_styling(latex_options = "hold_position")

#Utilize Boruta for feature ranking and selection
library(Boruta)

# Perform Boruta search
boruta_output <- Boruta(health_score ~ ., data=na.omit(df_num), doTrace=0, maxRuns = 1000)

#Get significant variables including tentatives
boruta_signif <- getSelectedAttributes(boruta_output, withTentative = TRUE)
#print(boruta_signif)

# Plot variable importance
plot(boruta_output, cex.axis=.7, las=2, xlab="", main="Variable Importance")

#Feature selection (address weak target correlation and multicollinearity)
select_df <- df_num[, c(1,4,6,8,11:14,16,18,23)]
#head(select_df) #verify

#Histograms for all variables
select_df %>%
    gather() %>%
    ggplot(aes(value)) +
        facet_wrap(~ key, scales = "free", ncol=4) +
        geom_histogram(bins=20,color="darkblue", fill="lightblue")

#Ensure reproducibility
set.seed(123)

#Train-test split data
dt = sort(sample(nrow(select_df), nrow(select_df)*.75))
train <- select_df[dt,]
test <- select_df[-dt,]
#dim(train) #2253 x 11
#dim(test) # 751 x 11

###LINEAR REGRESSION (baseline model): prior to outlier handling, normalization, feature engineering

model_1 <- lm(health_score ~., data = train)
summary(model_1) #R^2 = 0.6603, vars = 10


#Imputation / removal (address NAs): 0 missing values

#Normalization: 0 impact on R^2
norm_minmax <- function(x){(x- min(x, na.rm = TRUE)) /(max(x, na.rm = TRUE)-min(x, na.rm = TRUE))}
norm_df <- as.data.frame(lapply(train, norm_minmax))
#head(norm_df) #verify

#Outlier handling: reduced R^2
#cooksD <- cooks.distance(model_1)
```

```r
#influential <- as.numeric(names(cooksD)[(cooksD > (10 * mean(cooksD, na.rm = TRUE)))])
#train[influential,] #verify outliers - 20 rows
#out_df <- train[-influential,]

#Feature engineering

#combine datasets so we don't have to make features twice
train$dataset <- 'train'
test$dataset <- 'test'
final_df <- rbind(train, test)

#Creating new features (started with 1st, 3rd quartile values then adjusted)
##strong EQI and lots of sun
final_df$env_and_sun <- as.factor(ifelse(final_df$EQI > 0.90017 & final_df$Sun > 16350, 1, 0))
##high proportion college graduates and high income (3rd quartile)
final_df$college_and_income <- as.factor(ifelse(final_df$College > 20.0 & final_df$Income > 65000, 1, 0
##low binge drinking and high income (older, more responsible?)
final_df$lowDrink_and_HiIncome <- as.factor(ifelse(final_df$Bng < 32.5 & final_df$Income > 61629, 1, 0))
##high binge drinking and high income (expendable income, socially active)
final_df$HiDrink_and_HiIncome <- as.factor(ifelse(final_df$Bng > 35.0 & final_df$Income > 61629, 1, 0))
##high food insecurity and high mortality and less than HS
final_df$LowFood_HiDeath_LowEd <- as.factor(ifelse(final_df$FoodInsecurity > 15.7 & final_df$Mortality
##low income and less than HS and growth in unemployment
final_df$LoIncome_LowEd_HiUnemp <- as.factor(ifelse(final_df$Income < 30000 & final_df$UnemploymentChg
##big PopChg and low unemployment
final_df$HiPop_LoUnemp <- as.factor(ifelse(final_df$PopChg > 1595.5 & final_df$UnemploymentChg > -1, 1,

#Transformed (model 2): after handling outliers, normalization, feature engineering
train2 <- final_df %>% filter(dataset == 'train') %>% dplyr::select(-dataset)
test2 <- final_df %>% filter(dataset == 'test') %>% dplyr::select(-dataset)
dat <- final_df %>% dplyr::select(-dataset)
model_2 <- lm(health_score ~., data = train2)
#summary(model_2) #R^2 = 0.6735, vars = 17

###LINEAR REGRESSION (transformed model): features created, AIC optimized
#stepAIC(model_2)

model_3 <- lm(formula = health_score ~ Mortality + College + FoodInsecurity +
    LTHighSchool + Sun + EQI + Income + PopChg + Bng + UnemploymentChg +
    env_and_sun + lowDrink_and_HiIncome + HiDrink_and_HiIncome +
    LowFood_HiDeath_LowEd + LoIncome_LowEd_HiUnemp + HiPop_LoUnemp,
    data = train2)

#summary(model_3) #R^2 = 0.6734, vars = 16

# Predict and evaluate raw_lm model on training data
predictions = predict(model_3, newdata = train2)
eval_metrics(model_3, train2, predictions, target = 'health_score') #0.671, 0.0764

# Predict and evaluate raw_lm model on testing data
predictions = predict(model_3, newdata = test2)
eval_metrics(model_3, test2, predictions, target = 'health_score') #0.671, 0.0815
```

```r
## --- Model Building --- ##

###LINEAR REGRESSION (raw data)

#Train-test split data
dt2 = sort(sample(nrow(df_num), nrow(df_num)*.75))
train_raw <- df_num[dt2,]
test_raw <- df_num[-dt2,]
#dim(train_raw) #2253 x 23
#dim(test_raw) # 751 x 23

#Train raw_lm model:
raw_lm <- lm(health_score ~., data = train_raw)
#summary(raw_lm) #R^2 = 0.6808, vars = 22

# Predict and evaluate raw_lm model on training data
predictions = predict(raw_lm, newdata = train_raw)
eval_metrics(raw_lm, train_raw, predictions, target = 'health_score') #0.6678, 0.0759

# Predict and evaluate raw_lm model on testing data
predictions = predict(raw_lm, newdata = test_raw)
eval_metrics(raw_lm, test_raw, predictions, target = 'health_score') #0.6678, 0.0827

###LINEAR REGRESSION (raw data, AIC optimized)
#stepAIC(raw_lm)

#Train aic_raw_lm model:
aic_raw_lm <- lm(formula = health_score ~ Hvy + HvyPctChg + Bng + Mortality +
    HighSchool + SomeCollege + College + EQI + FoodInsecurity +
    Sun + Unemployment + UnemploymentChg + Poverty + Population +
    Births + Deaths + NetMig, data = train_raw)

#summary(aic_raw_lm) #R^2 = 0.6802, vars = 17

# Predict and evaluate aic_raw_lm model on training data
predictions = predict(aic_raw_lm, newdata = train_raw)
eval_metrics(aic_raw_lm, train_raw, predictions, target = 'health_score') #0.6671, 0.0761

# Predict and evaluate aic_raw_lm model on testing data
predictions = predict(aic_raw_lm, newdata = test_raw)
eval_metrics(aic_raw_lm, test_raw, predictions, target = 'health_score') #0.6671, 0.0828

###RIDGE REGRESSION (raw data)

#Specify column names
cr_raw = c('Hvy', 'HvyPctChg', 'BngPctChg', 'MortalityChg', 'HighSchool', 'SomeCollege', 'College', 'Un

#Generate dummy variables from data (if applicable)
dummies <- dummyVars(health_score ~ ., data = df_num[,cr_raw])
train_dummies = predict(dummies, newdata = train_raw[,cr_raw]) #2253 x 22
test_dummies = predict(dummies, newdata = test_raw[,cr_raw]) #751 x 22
print(dim(train_dummies)); print(dim(test_dummies))
```

```r
#Create numeric model matrices
x = as.matrix(train_dummies)
y_train = train_raw$health_score
x_test = as.matrix(test_dummies)
y_test = test_raw$health_score

lambdas <- 10^seq(2, -3, by = -.1) #specify lambda sequence

#Train model
raw_ridge_reg = glmnet(x, y_train, nlambda = 25, alpha = 0, family = 'gaussian', lambda = lambdas)
summary(raw_ridge_reg)

#Compute optimal lambda
raw_cv_ridge <- cv.glmnet(x, y_train, alpha = 0, lambda = lambdas)
ol <- raw_cv_ridge$lambda.min
ol #0.003162278

#Compute R^2 from true and predicted values
eval_results <- function(true, predicted, df) {
  SSE <- sum((predicted - true)^2)
  SST <- sum((true - mean(true))^2)
  R_square <- round((1 - SSE / SST),4)
  RMSE = round(sqrt(SSE/nrow(df)),4)

  # Model performance metrics
data.frame(RMSE = RMSE, Rsquare = R_square)
}

#Predict and evaluate raw_ridge_reg model on training data
predictions_train <- predict(raw_ridge_reg, s = ol, newx = x)
eval_results(y_train, predictions_train, train_raw) #RMSE: 0.0761, RSquare: 0.6734

#Predict and evaluate raw_ridge_reg model on testing data
predictions_test <- predict(raw_ridge_reg, s = ol, newx = x_test)
eval_results(y_test, predictions_test, test_raw) #RMSE: 0.0789, RSquare: 0.659
#Ridge regression lambda plot
plot(raw_ridge_reg)

###RIDGE REGRESSION (transformed data)

#Specify column names
cols_reg = c('Mortality', 'College', 'FoodInsecurity', 'LTHighSchool', 'Sun', 'EQI', 'Income', 'PopChg'

#Generate dummy variables from data (if applicable)
dummies <- dummyVars(health_score ~ ., data = dat[,cols_reg])
train_dummies2 = predict(dummies, newdata = train2[,cols_reg]) #2253 x 22
test_dummies2 = predict(dummies, newdata = test2[,cols_reg]) #751 x 22
print(dim(train_dummies2)); print(dim(test_dummies2))

#Create numeric model matrices
x2 = as.matrix(train_dummies2)
y_train2 = train2$health_score
x_test2 = as.matrix(test_dummies2)
```

```r
y_test2 = test2$health_score

#Train model
ridge_reg = glmnet(x2, y_train2, nlambda = 25, alpha = 0, family = 'gaussian', lambda = lambdas)
summary(ridge_reg)

#Compute optimal lambda
cv_ridge2 <- cv.glmnet(x2, y_train2, alpha = 0, lambda = lambdas)
ol2 <- cv_ridge2$lambda.min
ol2 #0.003162278

#Predict and evaluate ridge_reg model on training data
predictions_train2 <- predict(ridge_reg, s = ol2, newx = x2)
eval_results(y_train2, predictions_train2, train2) #RMSE: 0.0764, RSquare: 0.6732

#Predict and evaluate ridge_reg model on training data
predictions_test2 <- predict(ridge_reg, s = ol2, newx = x_test2)
eval_results(y_test2, predictions_test2, test2) #RMSE: 0.0814, RSquare: 0.632

###LASSO REGRESSION (raw data)

# Setting alpha = 1 implements lasso regression
lasso_reg <- cv.glmnet(x, y_train, alpha = 1, lambda = lambdas, standardize = TRUE, nfolds = 5)

#Compute optimal lambda
lambda_best <- lasso_reg$lambda.min
lambda_best #0.001

raw_lasso <- glmnet(x, y_train, alpha = 1, lambda = lambda_best, standardize = TRUE)

predictions_train <- predict(raw_lasso, s = lambda_best, newx = x)
eval_results(y_train, predictions_train, train_raw) #RMSE: 0.0767, Rsquare: 0.6681

predictions_test <- predict(raw_lasso, s = lambda_best, newx = x_test)
eval_results(y_test, predictions_test, test_raw) #RMSE: 0.0783, Rsquare: 0.6642

#Lasso regression lambda plots
op <- par(mfrow=c(1, 2))
plot(lasso_reg$glmnet.fit, "norm",   label=TRUE)
plot(lasso_reg$glmnet.fit, "lambda", label=TRUE)
par(op)

###LASSO REGRESSION (transformed data)

# Setting alpha = 1 implements lasso regression
lasso_reg2 <- cv.glmnet(x2, y_train2, alpha = 1, lambda = lambdas, standardize = TRUE, nfolds = 5)

#Compute optimal lambda
lambda_best2 <- lasso_reg2$lambda.min
lambda_best2 #0.001

lasso_model <- glmnet(x2, y_train2, alpha = 1, lambda = lambda_best2, standardize = TRUE)
```

```r
predictions_train2 <- predict(lasso_model, s = lambda_best2, newx = x2)
eval_results(y_train2, predictions_train2, train2) #RMSE: 0.0765, Rsquare: 0.6724

predictions_test2 <- predict(lasso_model, s = lambda_best2, newx = x_test2)
eval_results(y_test2, predictions_test2, test2) #RMSE: 0.0811, Rsquare: 0.6354

## --- Model Selection --- ##

#Create Kable table to succinctly summarize model optimization results
Model <- c('1', '2', '3', '4', '5', '6', '7')
Method <- c('Linear', 'Linear', 'Linear', 'Ridge', 'Ridge', 'Lasso', 'Lasso')
Data <- c('raw', 'raw(AIC)', 'transformed', 'raw', 'transformed', 'raw', 'transformed')
Var_Num <- c(22, 17, 16, 22, 16, 22, 16)
R2_train <- c(0.6678, 0.673, 0.671, 0.6734, 0.6732, 0.6681, 0.6724)
RMSE_train <- c(0.0759, 0.0759, 0.0764, 0.0761, 0.0764, 0.0767, 0.0765)
R2_test <- c(0.6678, 0.673, 0.71, 0.659, 0.632, 0.6642, 0.6354)
RMSE_test <- c(0.0827, 0.0842, 0.0815, 0.0789, 0.0814, 0.0783, 0.0811)

output <- cbind(Model, Method, Data, Var_Num, R2_train, RMSE_train, R2_test, RMSE_test)

output %>%
  kbl(caption = "Regression Model Comparison") %>%
  kable_minimal() %>%
  kable_styling(latex_options = "hold_position")
```