A Cluster-Based Approach to Fitting Regression Models

L. Austin Hadamuscin, James Hawkins, Tomiwa Omotesho, Deep Sagar Karki

STAT 6440: Data Mining

Dr. Shuchismita Sarkar

April 23, 2022

Abstract

Many have anxiety about dying early, others fear a long life of pain, but what factors lead to a long healthy life, and are those factors the same around the world? This paper attempts model healthy life expectancy using data retrieved from the World Health Organization (WHO). k-NN is used as a method of data imputation for missing data. Model Based and k-Means clustering were considered as clustering methods to split the countries into an unknown number of clusters. The ensemble method, random forest, was used to determine the important factors in each cluster and each cluster LASSO regression to model healthy life expectance at birth. Due to the relatively small size of the data set, leave-one-out cross validation (LOOCV) was used for any unknown variables that need data mined. Unsurprisingly, we found that there were two common important variables, the adult mortality rate, and infant deaths; with other important variables varying across the clusters.

Contents

1 Introduction

There are many aspects when it comes to having a healthy life. Determining the factors that lead to a healthy life is a tricky task, but if we were to take a broad view of health, one could simply quantify this by measuring similar individuals' lifespans. An easy and convenient way to group similar individuals is by country. Although most countries contain residents from many different ethnicities, social statuses, religions, etc., they share the same government policies and a general geographical area. In this analysis, we used clustering methods to group our data and explored the explanatory variables in each group to determine how important a role each plays in determining one's healthy life expectancy.

1.1 Objectives

- To inspect the efficacy of the features included in the data set for the improvement of average lifespan of the countries.
- To determine whether there are better ways to group countries other than developed and undeveloped for the purpose of modeling life expectancy.
- To build predictive models that emphasize potential factors which could significantly improve the life expectancy of a population.

2 Data Description

All data was retrieved from the World Health Organization's (WHO) website and consists of health data from 180 countries. Our variables include:

1. Healthy life expectancy (HALE) at birth (years)
2. Adult Mortality – mortality rate per 1000 people aged 15-60

3. Infant Deaths – Infant mortality rate (between birth and 11 months per 1000 live births)
4. Alcohol – Recorded alcohol consumption (ages 15+) per capita in liters of pure alcohol
5. Percentage Expenditure – Domestic general government health expenditure (GGHE-D) as a percentage of general government expenditure (GGE) (%)
6. Hepatitis B – Hepatitis B (HepB3) immunization coverage among 1-year-olds (%)
7. BMI – Mean body mass index trends among adults, crude (kg/m^2)
8. Polio – polio (Pol3) immunization coverage among 1-year-olds (%)
9. Total expenditure – Domestic general government health expenditure (GGHE-D) as a percentage of gross domestic product (GDP) (%)
10. Diphtheria – diphtheria tetanus toxoid and pertussis (DTP3) immunization coverage among 1-year-olds (%)
11. HIV/AIDS – deaths per 1000 live births with HIV/AIDS (0-4 years)
12. Thinness 5-19 years – prevalence of thinness among adolescents age 5 to 19, BMI < -2 standard deviations below the median (crude estimate) (%)

3 Methodology

3.1 Leave-One-Out Cross-Validation

The relatively small number of observations in our data allowed us the luxury of using leave-one-out cross validation (LOOCV) whenever we needed to optimize a parameter. LOOCV uses the following steps:

- partition all data into a training set except for one observation that will be used as our testing set;
- build a model using the training set;

- use this model to predict the response value in the testing set and calculate the mean square error (MSE);
- and repeat this process until every observation has been used in the testing set.

The test MSE is then calculated by taking the average of the MSE$_i$ values calculated in the LOOCV process. This process is repeated using different sets of parameters and the set of parameters with the lowest test MSE are our optimal parameters. To facilitate this process, we used the caret package (Kuhn, 2021), along with other packages in the statistical programming language R (R Core Team, 2021) (RStudio Team, 2021). Using LOOCV over other methods, such as k-fold cross validation, gives us a low bias estimate of test MSE because our ”training” set contains almost all the data; however, because substantial correlation exists between the MSE$_i$ from the overlapping in the training and testing sets, the variance of the estimate of test MSE is very high, meaning that any addition of new data could result in vastly different outcomes for our optimal parameters. We chose to accept this variance-bias trade-off because each observation is one country, so partitioning our data into a training, validation, and testing set would mean that whole countries would be excluded from the model building process which greatly reduces the applicability of our final model.

3.2 k-NN

We used k nearest neighbors (k-NN) to impute missing data. k-NN works by calculating the Euclidean distance between each observation in the data set, $\boldsymbol{x} = (x_1, x_2, \dots, x_p)$ and each new observation, $\boldsymbol{y} = (y_1, y_2, \dots, y_p)$: $d(\boldsymbol{x}, \boldsymbol{y}) = \sqrt{(y_1 - x_1)^2 + (y_2 - x_2)^2 + \cdots + \left(y_p - x_p\right)^2}$, where p represents the number of predictors. In classification k-NN, the new value takes the class of the majority among its k closest observations based on Euclidean distance. In the case of ties, there are several methods used to classify the new observations such as using the majority class among all observations. Our response variable is continuous, so we used regression k-NN where new observations are assigned the average response of the k closest observations. We used this concept to impute our missing data using our explanatory variables. We data mined the optimal k by utilizing the train function from the caret R package (Kuhn, 2021) and the knn method from the FNN R package (Beygelzimer, et al., 2019) to perform LOOCV as described in section 3.1. While doing initial research into k-NN data imputation, we found the function knnImputation from the archived R package DMwR (Torgo, 2010). We modified the code of this function to estimate the missing values for each observation, using the mean of the present values from their nearest neighbors.

3.3 k-Means Clustering

We used a k-means approach to cluster the countries using our explanatory variables. The k-means clustering algorithm is fairly simple and can be easily explained:

- randomly choose k points in p-dimensional space to use as the initial centroids;
- calculate the Euclidean distance between the observations and the centroids and assign each observation to its closest centroid;
- recompute new centroids by calculating the center of each group that was created in the previous step. Repeat the previous step.
- The algorithm stops when there is no change to the cluster assignments.

We can measure how well a k-means algorithm performs by calculating its within sum of squares (wss) which is a measure of the density of observations in the clusters and is defined as:

$$wss = \sum_{k=1}^{K} \sum_{i \in S_k} \sum_{j=1}^{p} \left(x_{ij} - \bar{x}_{kj}\right)^2,$$

where S_k is the set of observations in cluster k. The performance of a k-means clustering algorithm is highly dependent on the initialization of the centroids created in the first step of the algorithm so to combat this, we randomly generated a set 1000 seeds from a uniform(0,900000) distribution, using the ceiling function (R Core Team, 2021) to round to the next highest whole number. We used these seeds to run 1000 iterations of the algorithm, choosing the seed that produced the lowest wss. Using the same 1000 seeds, this process was repeated for $k \in [1,20]$ and the model that resulted in the lowest wss without the increase in k resulting in a negligibly lower wss was chosen.

3.4 Model-Based Clustering

We also took a model-based approach to clustering our data using our explanatory variables. Model-based clustering uses finite mixture models which are a linear combination of weighted density functions:

$$g(\boldsymbol{x}, \boldsymbol{\Theta}) = \sum_{k=1}^{K} \alpha_k f_k(\boldsymbol{x}; \boldsymbol{\Theta_k});$$

where α_k are the weights, $0 \leq \alpha_k \leq 1$, $\sum_{k=1}^{K} \alpha_k = 1$, and K is the mixture order, or number of components, $f_k(\boldsymbol{x}; \boldsymbol{\Theta}_k)$ that use the Expected-Maximization (EM) iterative algorithm to estimate $\boldsymbol{\Theta}_k$. Like the k-means clustering algorithm, model-based clustering algorithms are highly dependent on initialization so many models must be run to get the optimal assignment of clusters. To determine what set of clusters are the best, a modified version of the likelihood, Bayesian Information Criterion (BIC), is used:

$$BIC = -2\log\left(\hat{L}\right) + p\log(n);$$

where $\hat{L}$ is the maximum likelihood and p is the number of parameters to be estimated. We used the Mclust function from the mclust R package (Scrucca, Fop, Murphy, & Raftery, 2016) to perform the clustering. This package automatically runs the algorithm at different initializations and chooses the best model and returns its negative BIC. We ran the function using $K \in [1,20]$ and chose the model with the maximum negative BIC without the increase in K resulting in a negligibly higher negative BIC. We then compared this model, visually, with the model we created using k-means and chose the one that we felt created the best clusters.

3.5 Random Forest

Random forest is an ensemble technique that was first proposed by Leo Breiman from the University of California in 2001 (Parmar, Katariya, & Patel, 2018). Random forests consist of multiple decision trees created from a sample of variables whose results are averaged to obtain a final outcome and can be used in both regression and classification problems. This technique help reduce the bias of the final model without reducing its variance. The following are the steps used in random forest:

- Fix a proper value of predictors usually labelled as m.
- Select a new subset of predictors θ_k from the whole set of predictors depending on m
- Use θ_k to create a decision tree.
- Choose a new θ_k and repeat the process above until the algorithm travels through all the feature subsets.

The randomness incorporated in the model building process, such as the selection of sample subsets and feature subsets, guarantees the independence of each decision tree. The influence of m is highly noted on the performance of random forests (Parmar, Katariya, & Patel, 2018). With the increase in the value of m, the correlation between each tree in the training model can be decreased to make the trees independent.

We applied random forests to each cluster's explanatory variables and extracted variable importance. The variable importance is one of the important criteria in selecting key features on the model as it gives us the ability to rank the features by their predictive strength. It also tells us the most likely variables used for the split during the formation of decision trees in our model.

Since we had 11 predictors, we set the range of m predictors to be $m \in (2,10)$ and used LOOCV on each cluster to decide the optimal number of m predictors based on the residual mean square error (RMSE). The rf method from the R package, caret (Kuhn, 2021), was used to build the random forest, where we were able to determine our predictors' importance.

3.6 Linear Regression

Random forests do not yield a single, easily interpretable model; therefore, we decided it necessary to create a multiple linear regression model that we could use for both interpretation and prediction. The model assumptions for multiple linear regression are:

- Errors are normally distributed.
- Constant variance in the residuals.
- The relationship between predictors and response should be linear.
- The observations should be independent of each other.

We often encounter the problem of model misspecification with multiple linear regression. The model is prone to using extraneous variables that lead to overfitting. It is also the case that the model lacks some important variables. To avoid this, we used regularized regression techniques for variable selection. In a regularized regression technique, the regression coefficients are estimated by minimizing SSE+P, where SSE represents the sum of square deviations and P is the penalty factor. In regularized regression the estimated effects are slowly shrunken towards zero. As $\lambda \rightarrow 0$, the penalty factor P becomes larger and that forces the regression coefficients to become 0. The regularized regression has two types:

(i) Ridge regression: minimizes SSE+$\lambda \sum_{j=1}^{k} \beta_j^2$

(ii) Lasso regression. Minimizes SSE+$\lambda \sum_{j=1}^{k} |\beta_j|$

λ is called the tuning parameter, and as the tuning parameter becomes very large, the penalty factor becomes large that forces the regression coefficients to become zero. This process results in reducing the variance, but bias is compromised unlike in ordinary least square regression.

In the R package, glmnet (Friedman, Hastie , & Tibshirani, 2010), is used for regularized regression. Regularized regression requires the features to be standardized but the glmnet package does the standardization for us. The α parameter in the package is used to specify us whether we were using lasso regression (α=1) or ridge regression (α=0). We decided to use lasso regression, setting α=1. λ was then data mined based on the minimum.

Once, the variables are selected the model is fitted with the appropriate regressors. It is also important that our model does not suffer from potential multicollinearity. So, once the model is fitted, we check the multicollinearity in the model by using the variance inflation factor (VIF) from the car package (Fox & Weisberg, 2019). The VIF values higher than 4 signifies that the model suffers from moderate multicollinearity, whereas the VIF higher than 8 signifies that the model suffers from severe multicollinearity. If the problem of multicollinearity exists, then we will drop the highly correlated variables to get rid of the multicollinearity from the existing model. The model without multicollinearity is used for prediction purposes. The significance of the regressors in the model will be compared with the significance level of α=0.05.

4 Results

4.1 k-Means

Following our methods in section 3.3, we ran the k-means algorithm using various values of k using the same set of 1000 seeds for each k. The results are displayed in figure 1.

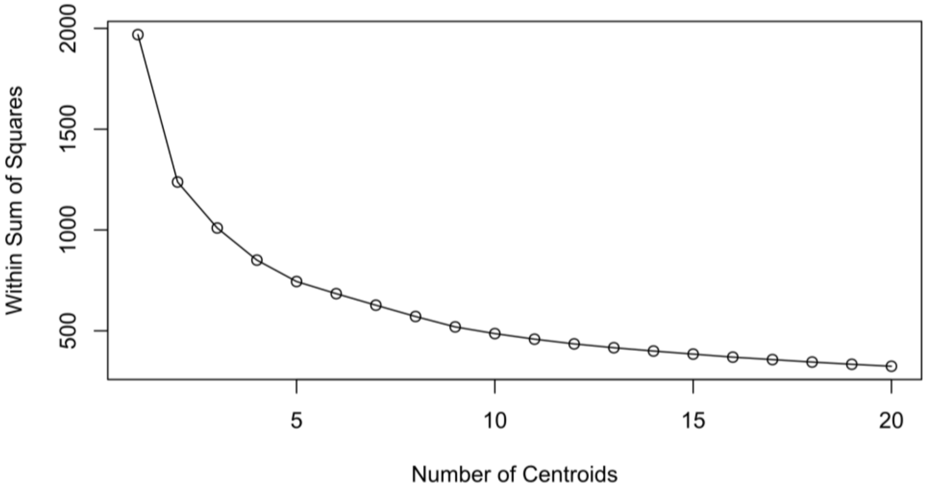


Figure 1: Within sum of squares for K-Means clustering

We can see that there is not a distinct “elbow” in the graph that can be chosen as our optimal k other than a slight bend at k = 2, which results in the map figure 2.

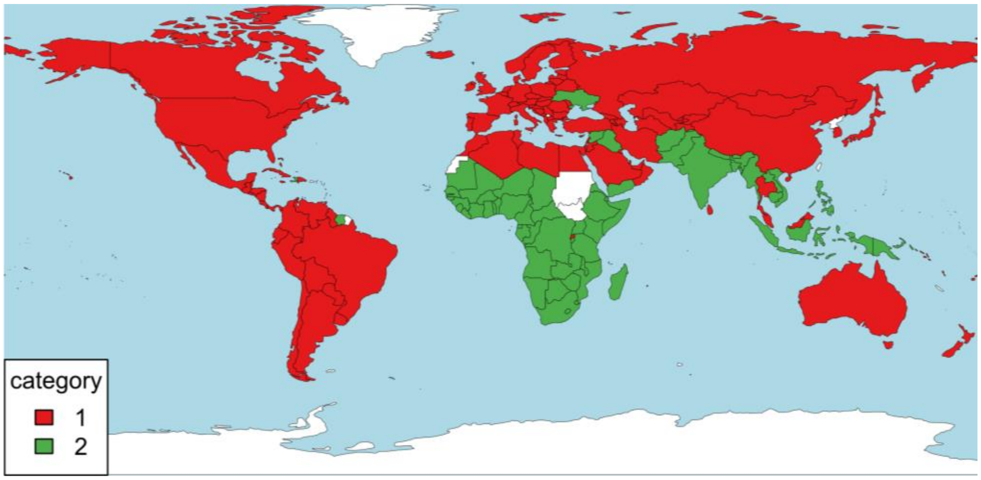


Figure 2: World map showing K-Means clusters at $k = 2$

The algorithm did a decent job at clustering the countries; however, there are issues that come with grouping the countries in such a broad manner. For example, less economically developed countries such as Turkey and Ecuador are grouped with more economically developed countries such as Germany and Australia. We looked at $k = 4$ to help break up some of these oddities. We chose $k = 4$ because reduction in the within sum of squares beyond this point was negligible. The resulting clusters can be seen in section 4.3 or a larger version can be accessed in appendix A. We can see that the k-means algorithm has done a good job of splitting up the African continent. The more economically developed North African countries are grouped with countries like Russia and China, while the rest of Africa was grouped into two. Likewise, we noticed Japan and South Korea were grouped with countries like the America and Australia which seemed very reasonable.

4.2 Model-Based

Using the methods described in section 3.4 we computed the computed the negative BIC of the model-based clustering algorithms as various values of K. The results are shown in figure 3.

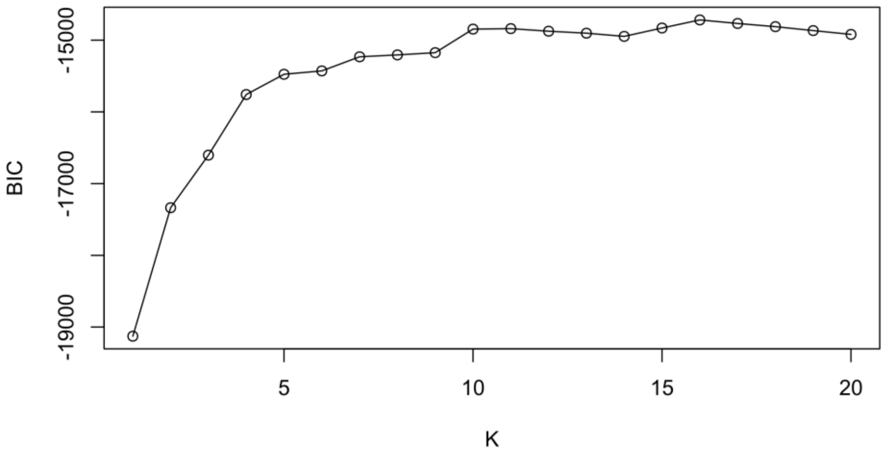


Figure 3: BIC for Model-Based clustering

As with the k-means model there is not a clear, reasonable optimal number of components, so we will be choosing 4 components for this model as the gain in negative BIC from adding more components is negligible. The resulting clusters can be seen in section 4.3 or a larger version can be accessed in appendix B. We can see that the model-based clustering did well but there are some problems, specifically in Asia where Japan is grouped with India and South Korea is grouped with Russia.

4.3 K-Means vs Model-Based

We believe that, visually, the 4-means clustering method did a better job splitting up Africa and Eastern Asia, specifically with South Korea and Japan in Asia, and South Africa and Somalia in Africa, as seen in the maps below. This is backed up by the means chart below. For example, Japan should not be in the group with countries that have the lowest Hepatitis B and Diphtheria vaccination rates. For this reason, we decided to choose the k-means clustering model.

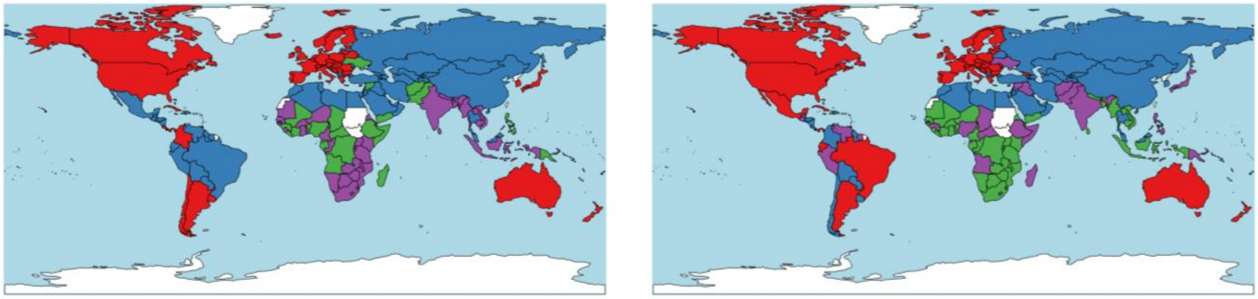


Figure 4: World map showing K-Means vs Model-Based clusters at $k = 4$

4.4 Random Forest

Table 1 shows each cluster’s optimal number of m and the corresponding RMSE from LOOCV, while table 2 shows the variable importance rank for each cluster. The variable importance plots for each of the clusters can be seen from appendices C through F.

Table 1: Optimal “m” tries and RMSE for the clusters

Cluster	Optimal “m” tries	RMSE
1	10	1.017949
2	6	2.400911
3	10	2.003527
4	8	2.451414

Table 2: Variable importance rank for the clusters

Rank	Cluster1	Cluster 2	Cluster 3	Cluster 4
1	Adult Mortality	Infant Mortality	Infant Mortality	Adult Mortality
2	Thinness (5-19)	Adult Mortality	Adult Mortality	HIV
3	Percent Expenditure	Thinness (5-19)	HIV	Infant Mortality
4	Thinness (5-19)	Total Expenditure	Hepatitis B	Thinness (5-19)
5	BMI	Diphtheria	Diphtheria	Total Expenditure
6	Infant Mortality	BMI	Percent Expenditure	Percent Expenditure
7	Alcohol	Polio	Alcohol	Polio
8	Total Expenditure	Alcohol	BMI	BMI
9	Hepatitis B	Percent Expenditure	Polio	Hepatitis B
10	Diphtheria	Hepatitis B	Total Expenditure	Diphtheria
11	HIV	HIV	Thinness (5-19)	Alcohol

4.5 Regression

4.5.1 Transformations

To satisfy the model assumption for linearity between our regressors and our explanatory variables, transformations were made as shown in table 3. Appendices G through R show the linear plots before and after transformation for each cluster. For cluster's 1 and 2, we decided to drop the HIV variable because the linearity assumption was not satisfied, as most of the observations had a value of 0.

Table 3: Variable transformations for the clusters

Cluster1	**Cluster 2**	**Cluster 3**	**Cluster 4**
log(Alcohol + 0.5)	log(Alcohol + 0.5)	log(Alcohol + 0.5)	log(Alcohol + 0.5)
Hepatitis B^5	Hepatitis B^5	Hepatitis B^5	Hepatitis B^5
log(Infant Mortality)	log(Infant Mortality)	log(Infant Mortality)	log(Total Expenditure)
log(Percent Expenditure)	log(Percent Expenditure)	log(Percent Expenditure)	
	$\sqrt{\text{Thinness } (5-19)}$	log(Total Expenditure)	
	$\sqrt{\text{Total Expenditure}}$		

4.5.2 Lasso Regression

After completing the transformations for the clusters, we performed lasso regression to eliminate multicollinearity in our final regression models. Tables 4 and 5 show the regressors whose coefficients were shrunk to zero and the significant regressors, respectively. The summaries for the final regression models for each cluster can be found in appendices S through V.

Table 4: Predictors whose coefficients were shrunk to zero

Cluster 1	**Cluster 2**	**Cluster 3**	**Cluster 4**
Diphtheria	Diphtheria	BMI	Diphtheria
	BMI	Polio	log(Alcohol)
	HepB^5	log(Percent Expenditure)	Polio
		Thinness (5-19)	Thinness (5-19)

	Total Expenditure

Table 5: Significant Predictors for the Clusters

Cluster 1	**Cluster 2**	**Cluster 3**	**Cluster 4**
Adult Mortality	Adult Mortality	log(Infant Mortality)	Adult Mortality
BMI	log(Alcohol)		HIV
Polio	log(Percent Expenditure)		
log(Percent Expenditure)	log(Infant Mortality)		
Thinness (5-19)			
Total Expenditure			

5 Conclusion

Based on the data of 2014 we implemented different algorithms to extract some useful information to meet our objectives. The implementation of the clustering technique was to find suggestive models for the countries depending on the necessities of the improvement on certain features. Countries belonging to cluster 1 clearly were more homogenous than countries belonging to the rest of the clusters as they were inclined to share similar features. This commonality potentially reflected similar problems as well. We noticed that all the clusters shared “mortality” as the crucial factor. It is natural to think that mortality and life expectancy are in inverse relation to each other. When we considered other factors for the countries, we found some strong features that needed to be addressed within each cluster separately.

Countries belonging to cluster 1 and cluster 2 were mostly countries with strong socio-economic status. So, naturally, the problems affecting life expectancy in these countries are different than the countries belonging to clusters 3 and 4, which represent the underprivileged countries. For instance, we noticed the risk of HIV is not on par with the risk in developed countries. We also noticed BMI and Alcohol were significant factors that could affect the life expectancy of the population in developed countries. However, they were not significant in underdeveloped countries. These differences strongly argue that the improvement of life expectancy in developed vs underdeveloped countries needs different strong priorities. Local authorities, government bodies, and international organizations should draw attention to the stated factors above for the betterment of their populations.

6 Limitation and future work

Nothing. This is a perfect project. We have solved world suffering.

We believe the methodology used for this project is ideal with no significant improvement required.

7 References

Beygelzimer, A., Kakadet, S., Langford, J., Arya, S., Mount, D., & Li, S. (2019). FNN: Fast Nearest Neighbor Search Algorithms and Applications. *R package version 1.1.3*. Retrieved from https://CRAN.R-project.org/package=FNN

Fox, J., & Weisberg, S. (2019). *An R Companion to Applied Regression* (Third ed.). Thousand Oaks, CA: Sage. Retrieved from https://socialsciences.mcmaster.ca/jfox/Books/Companion/

Friedman, J., Hastie , T., & Tibshirani, R. (2010). Regularization Paths for Generalized Linear Models via Coordinate Descent. *Journal of Statistical Software, 33*(1), 1-22. Retrieved from https://www.jstatsoft.org/v33/i01/

Kuhn, M. (2021). caret: Classification and Regression Training. R package version 6.0-90. Retrieved from https://CRAN.R-project.org/package=caret

Parmar, A., Katariya, R., & Patel, V. (2018). A review on random forest: An ensemble classifier. In *International Conference on Intelligent Data Communication Technologies and Internet of Things* (pp. 758-763). Springer. doi:10.1007/978-3-030-03146-6_86

R Core Team. (2021). R: A Language and Environment for Statistical Computing. Vienna, Austria: R Foundation for Statistical Computing. Retrieved from https://www.R-project.org/

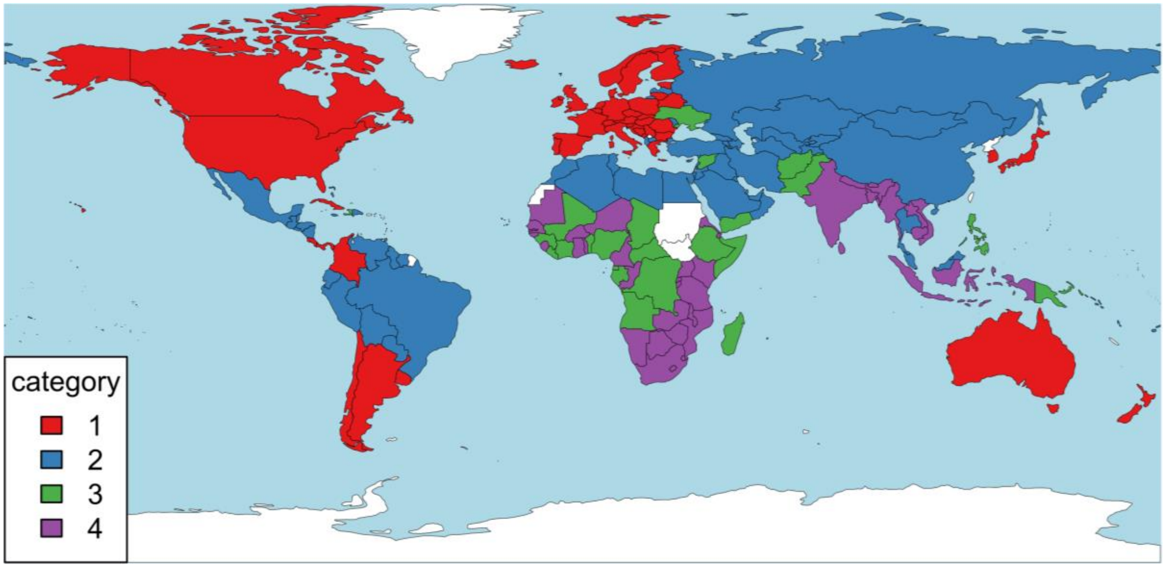
RStudio Team. (2021). RStudio: Integrated Development Environment for R. Boston, MA. Retrieved from http://www.rstudio.com/

Scrucca, L., Fop, M., Murphy, T. B., & Raftery, A. E. (2016). mclust 5: clustering, classification and density estimation using Gaussian finite mixture models. *The R Journal, 8*(1), 289-317. doi:https://doi.org/10.32614/RJ-2016-021

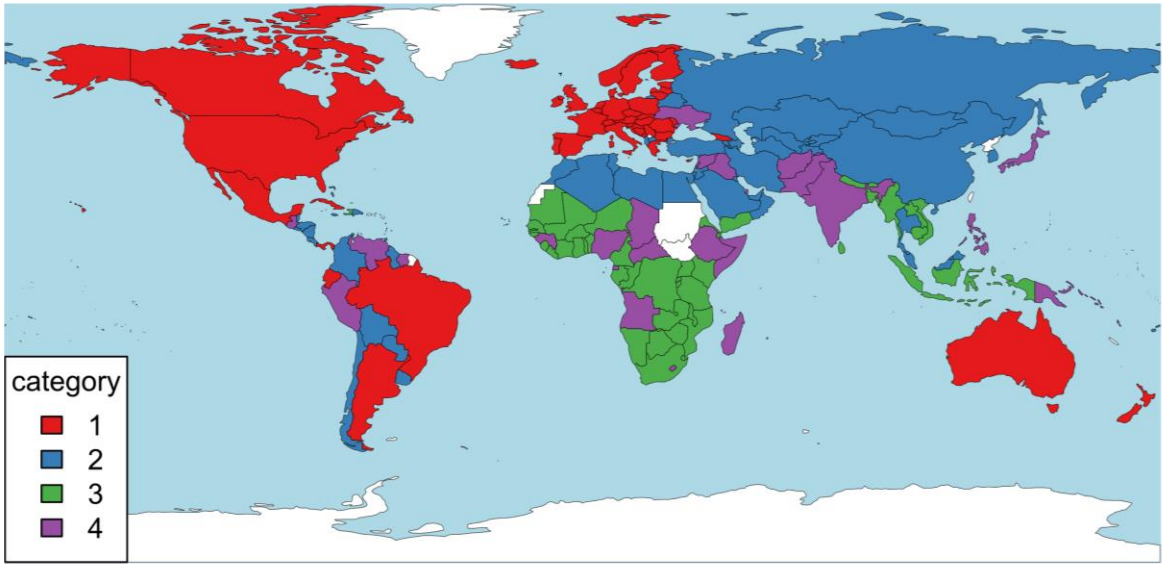
Torgo, L. (2010). Data Mining with R, learning with case studies. Chapman and Hall/CRC. Retrieved from http://www.dcc.fc.up.pt/~ltorgo/DataMiningWithR

Appendices

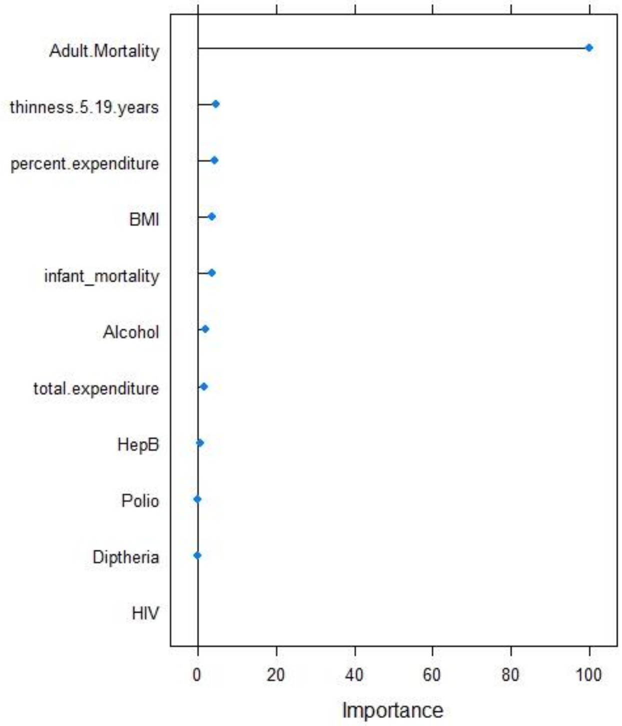
Appendix A: World map showing K-Means clusters at k = 4



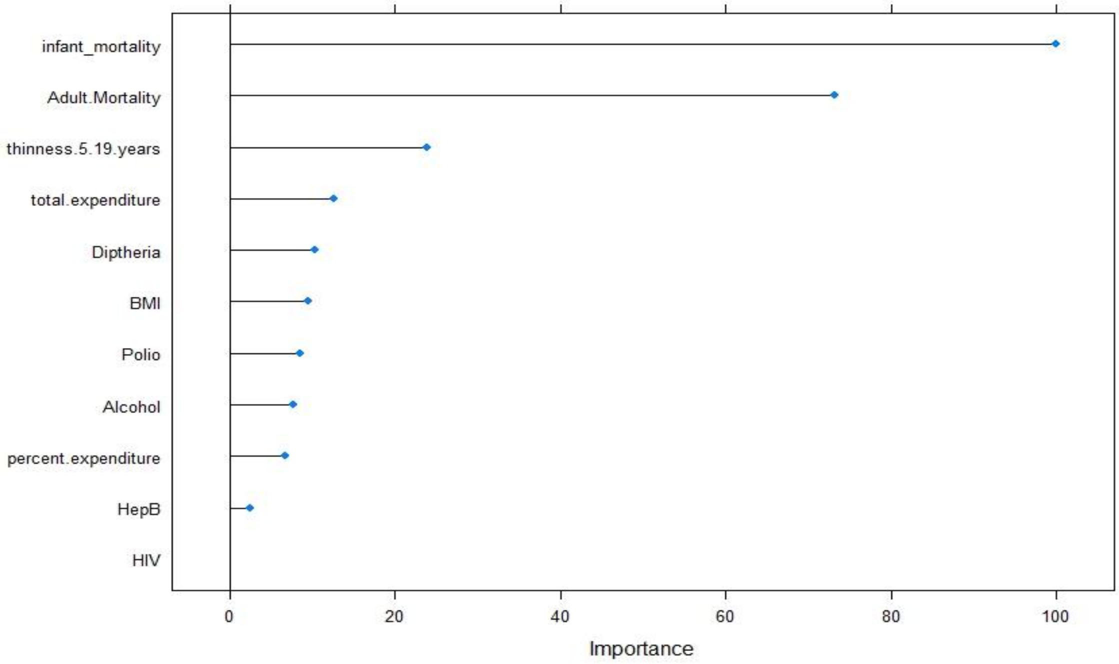
Appendix B: World map showing Model-Based clusters at k = 4



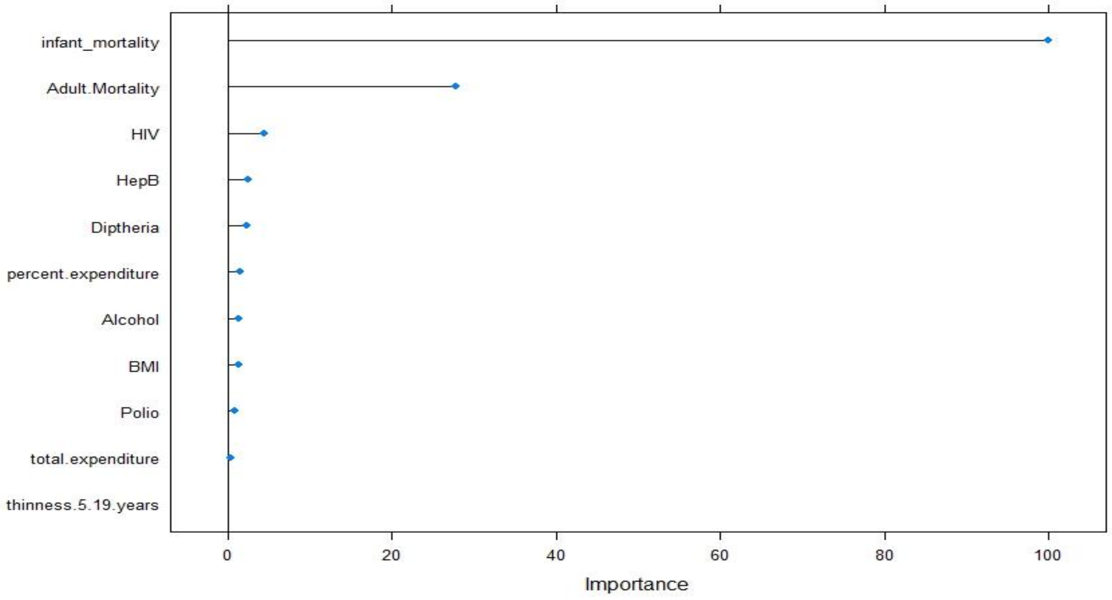
Appendix C: Variable importance for cluster 1



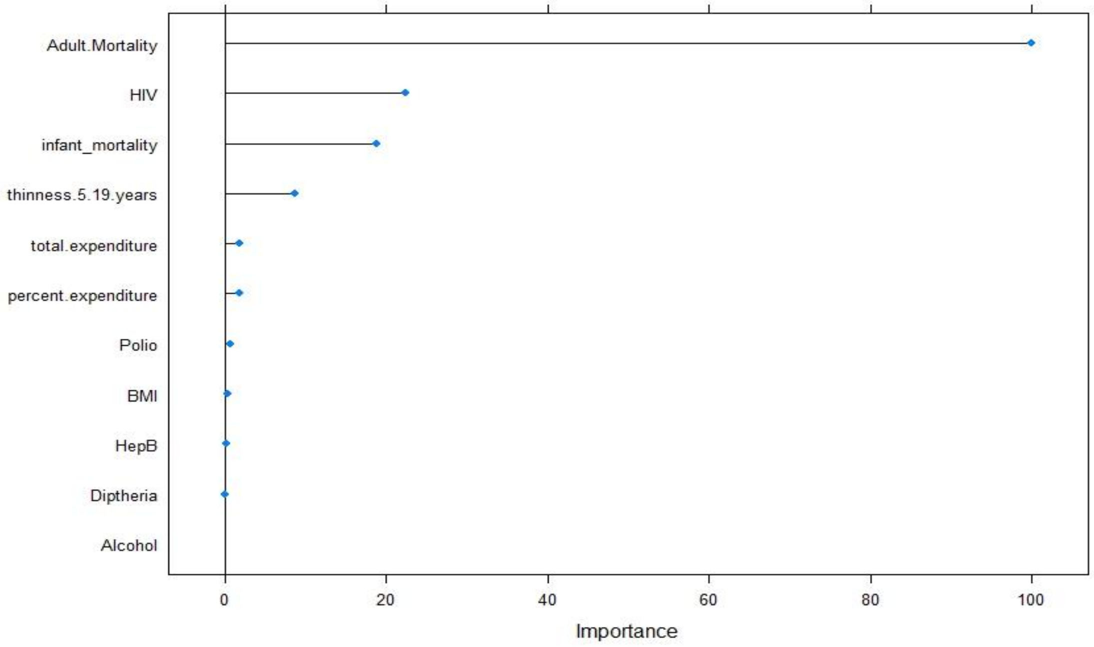
Appendix D: Variable importance for cluster 2



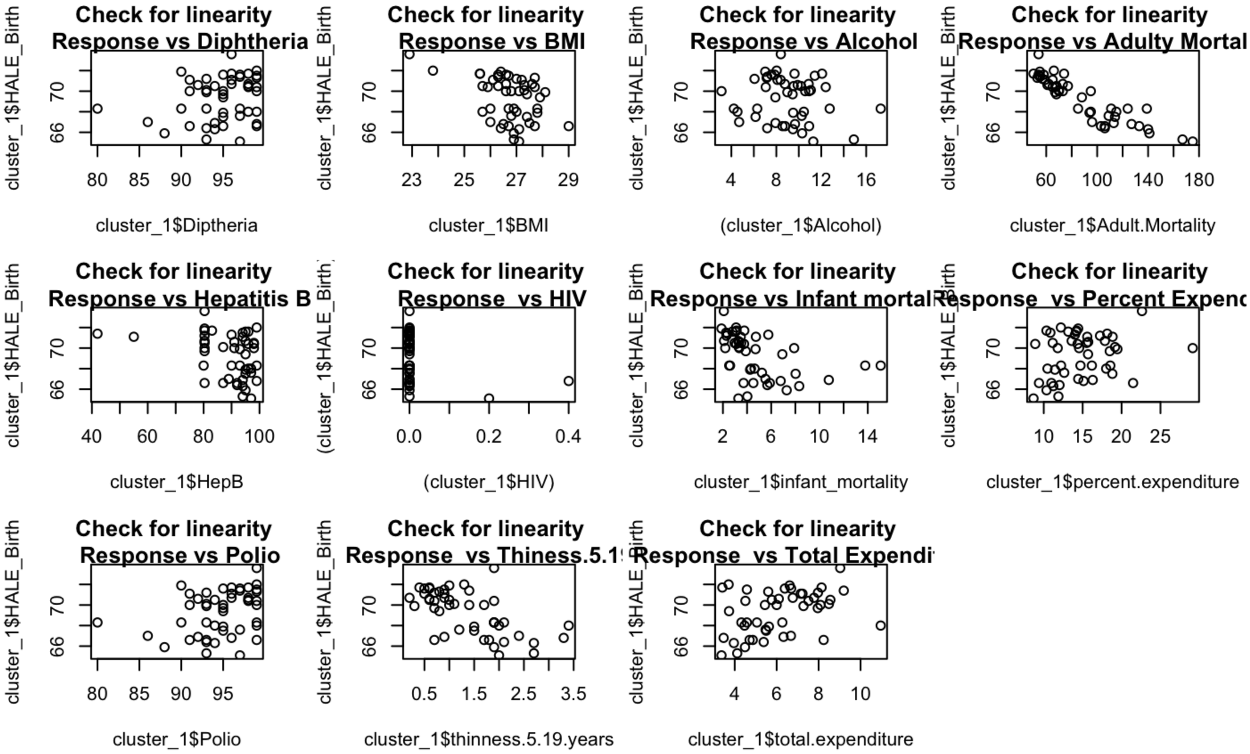
Appendix E: Variable importance for cluster 3



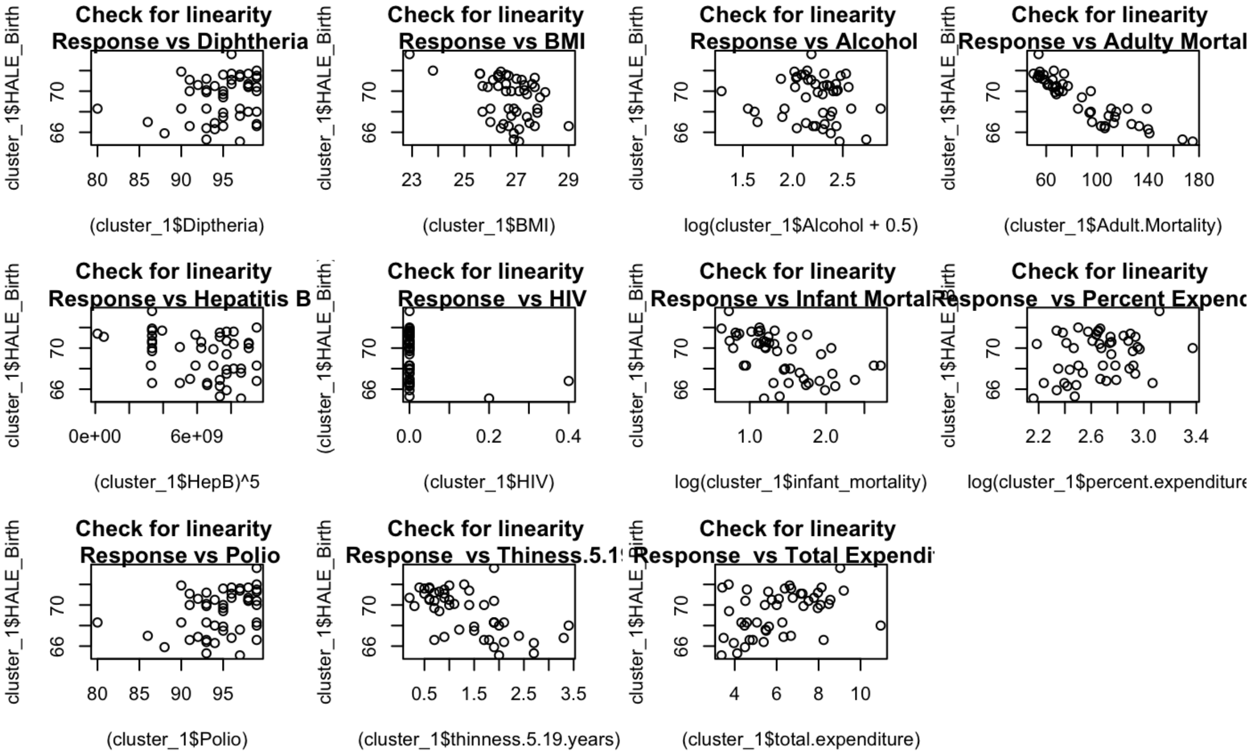
Appendix F: Variable importance for cluster 4



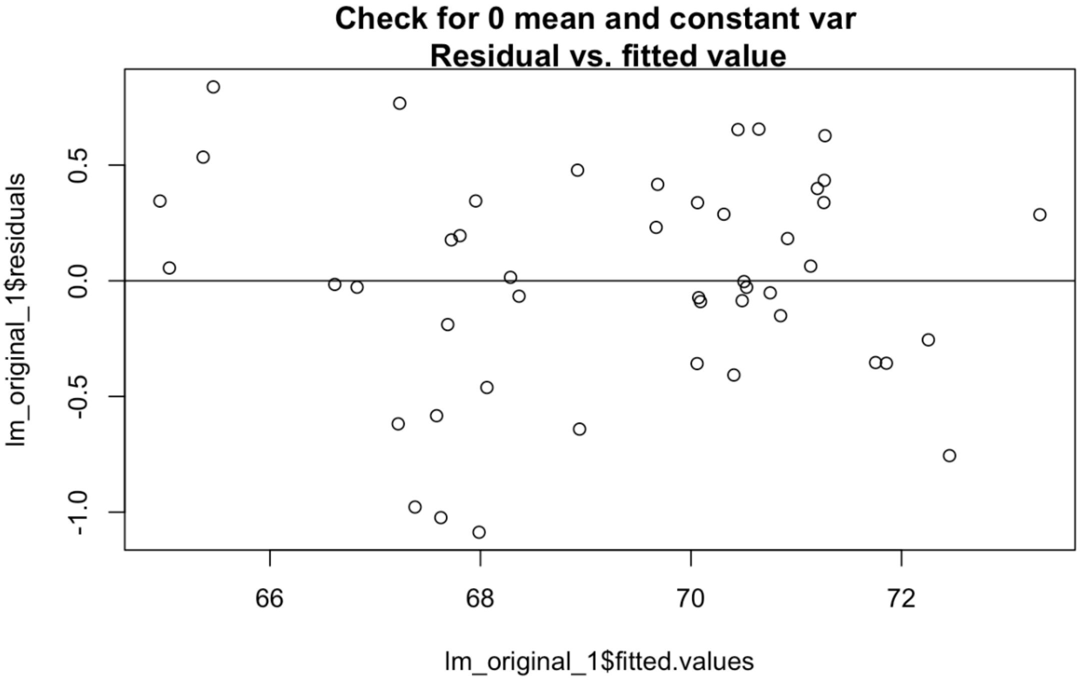
Appendix G: Linearity plots for regressor variables before transformation (cluster 1)



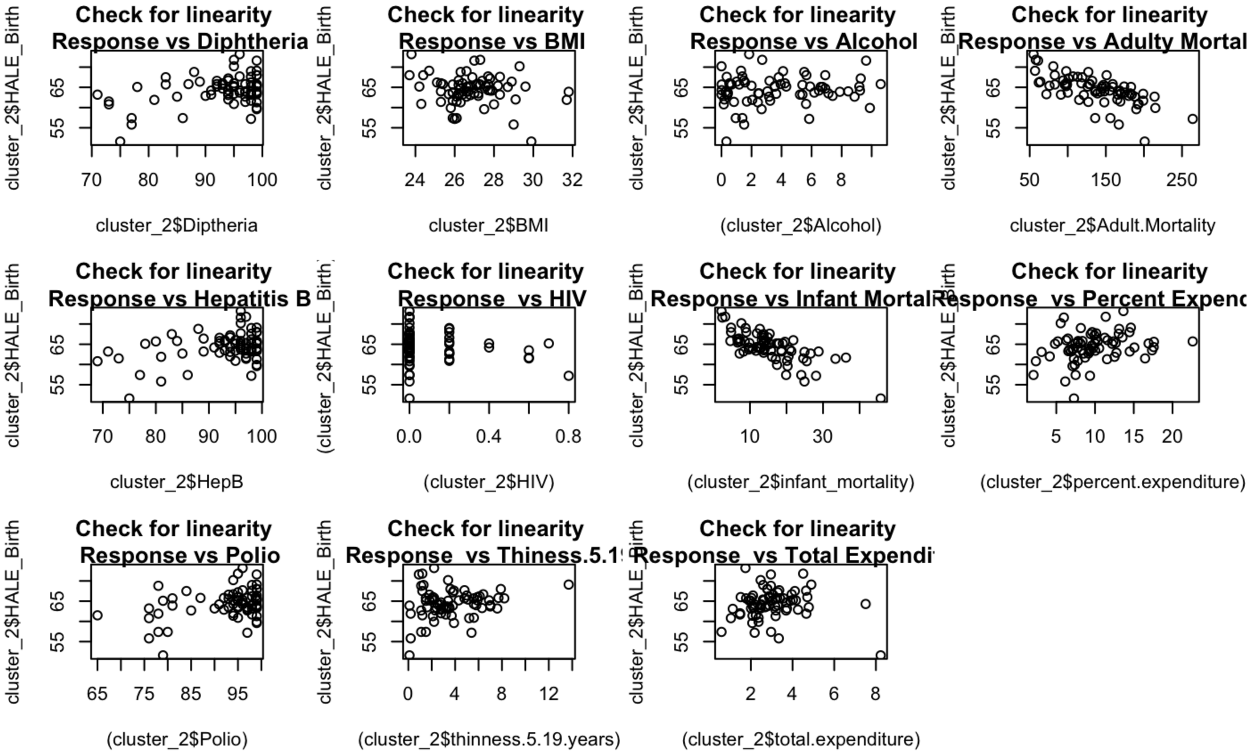
Appendix H: Linearity plots for regressor variables after transformation (cluster 1)



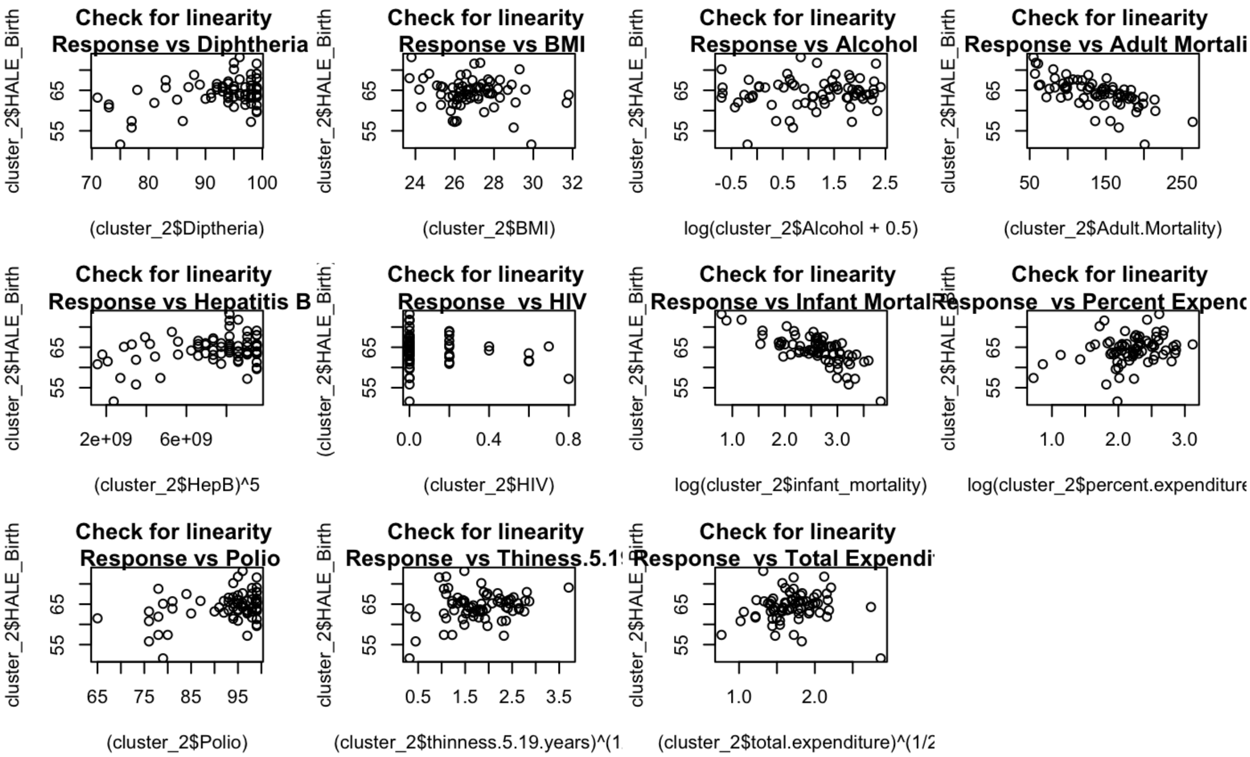
Appendix I: Residuals vs fitted values plot (cluster 1)



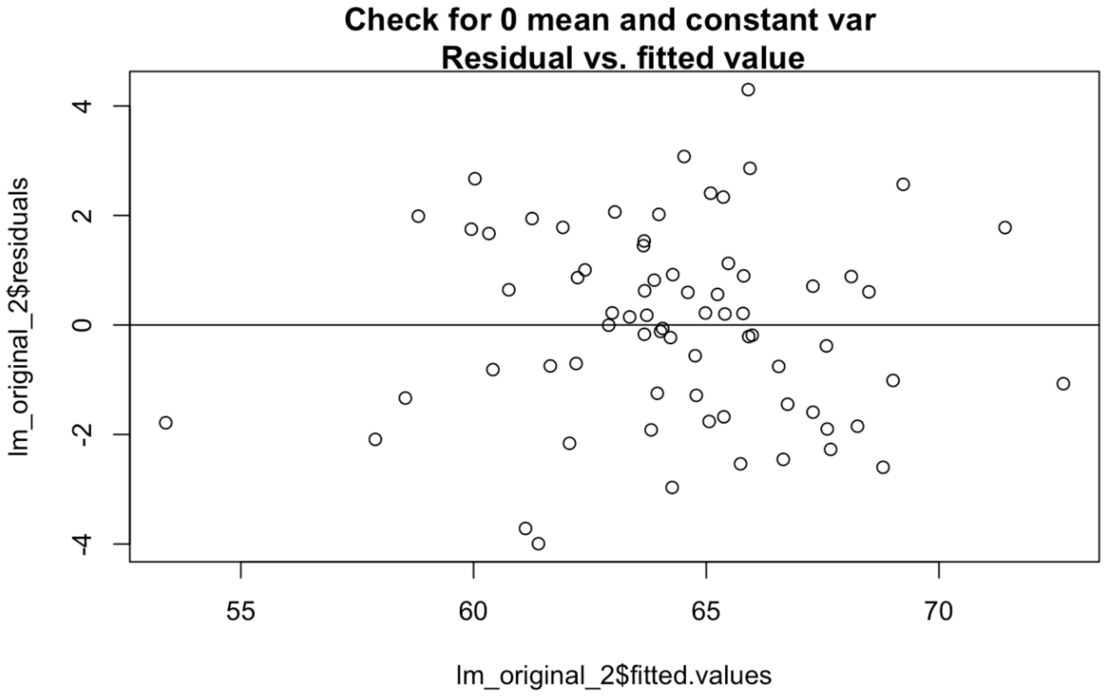
Appendix J: Linearity plots for regressor variables before transformation (cluster 2)



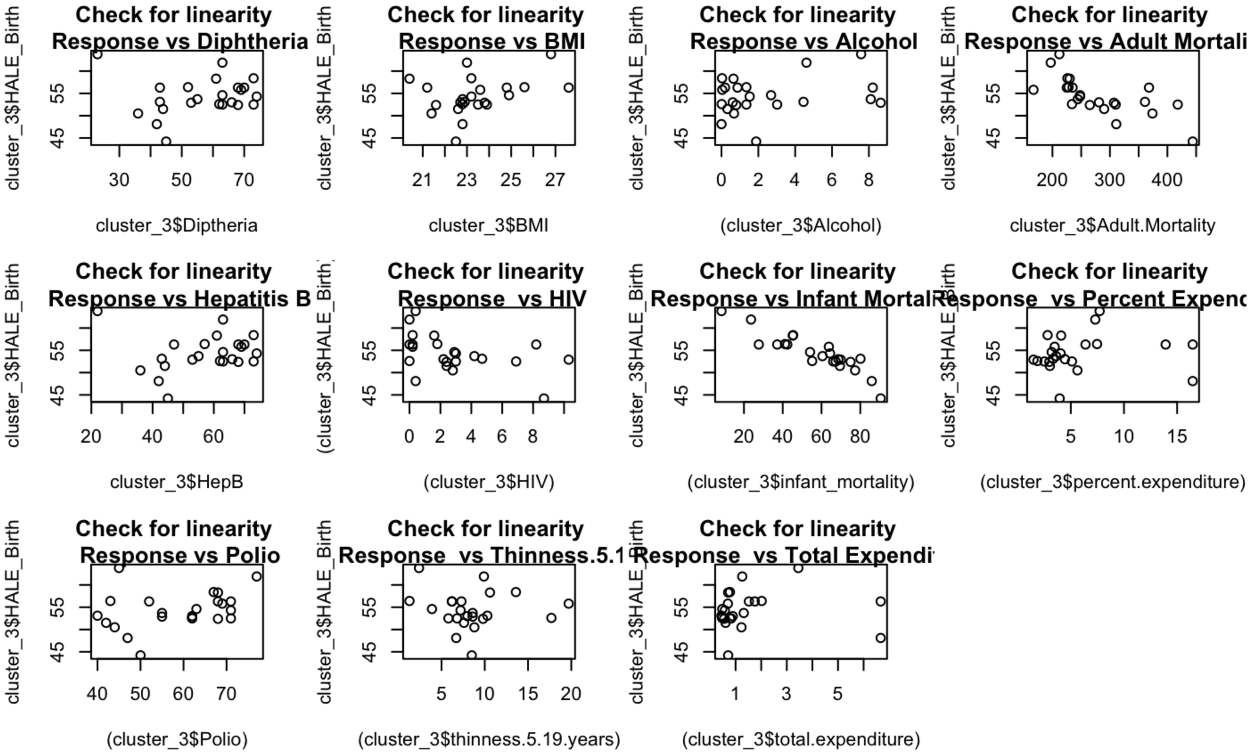
Appendix K: Linearity plots for regressor variables after transformation (cluster 2)



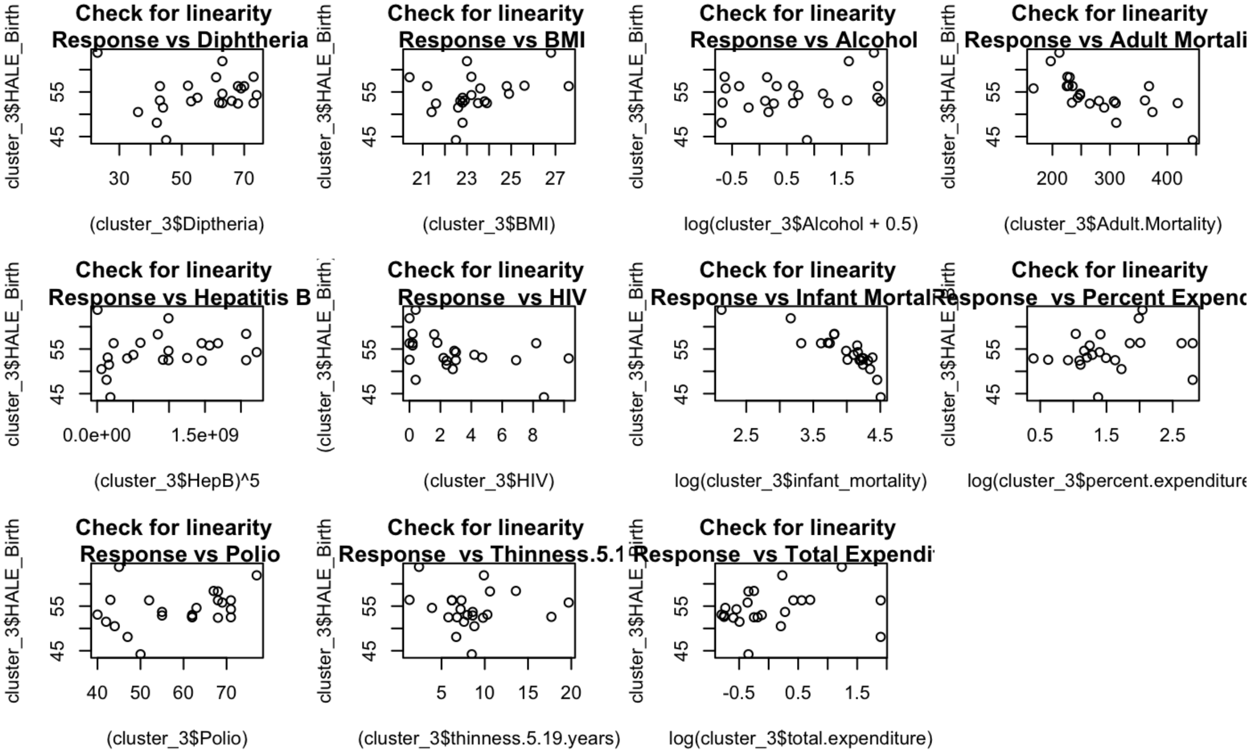
Appendix L: Residuals vs fitted values plot (cluster 2)



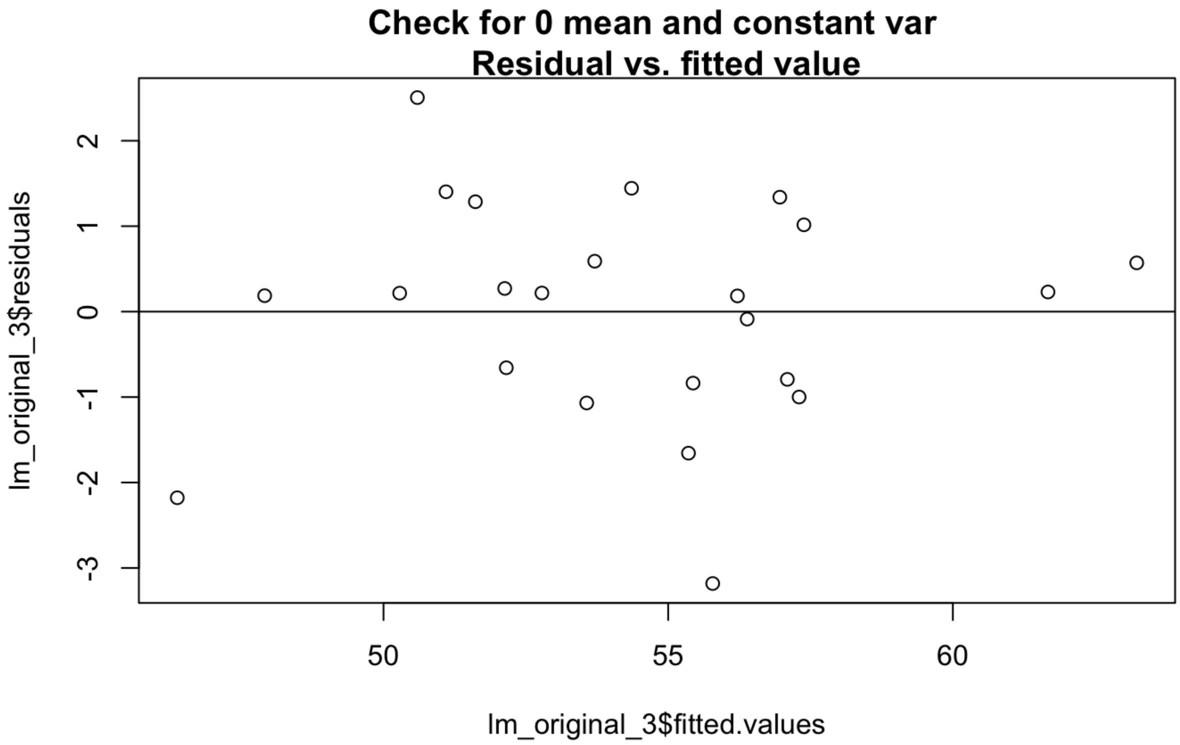
Appendix M: Linearity plots for regressor variables before transformation (cluster 3)



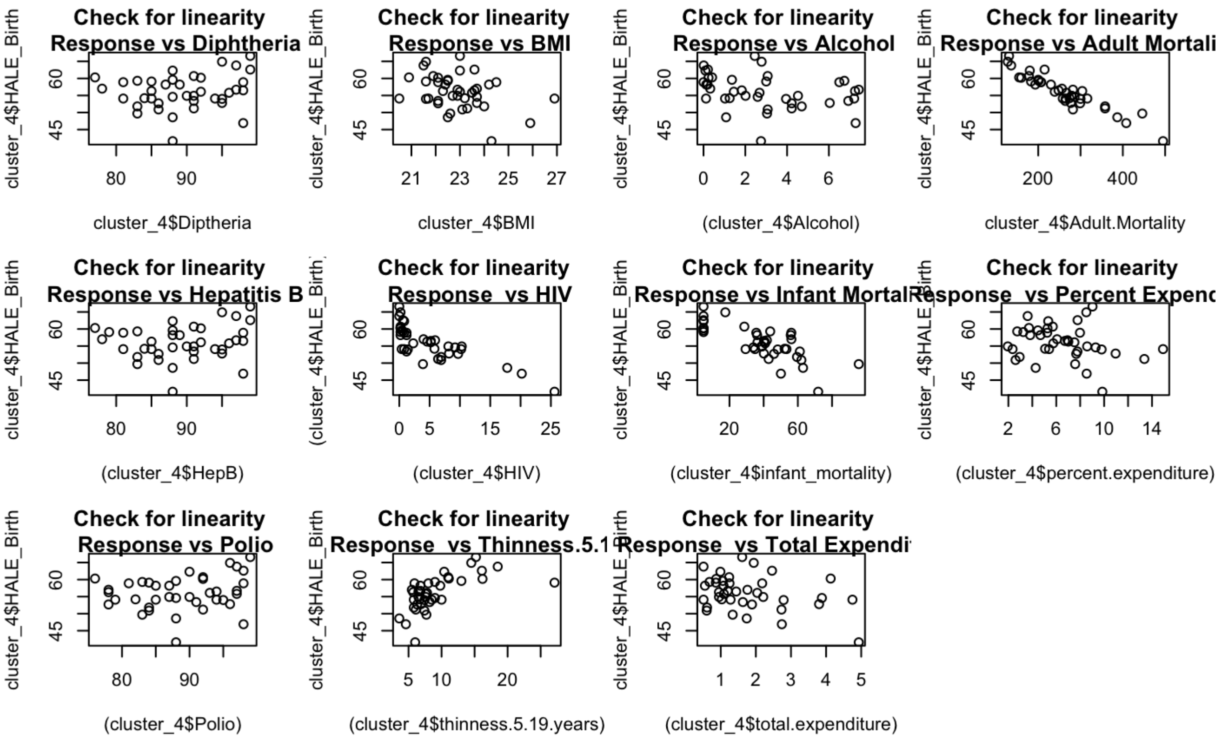
Appendix N: Linearity plots for regressor variables after transformation (cluster 3)



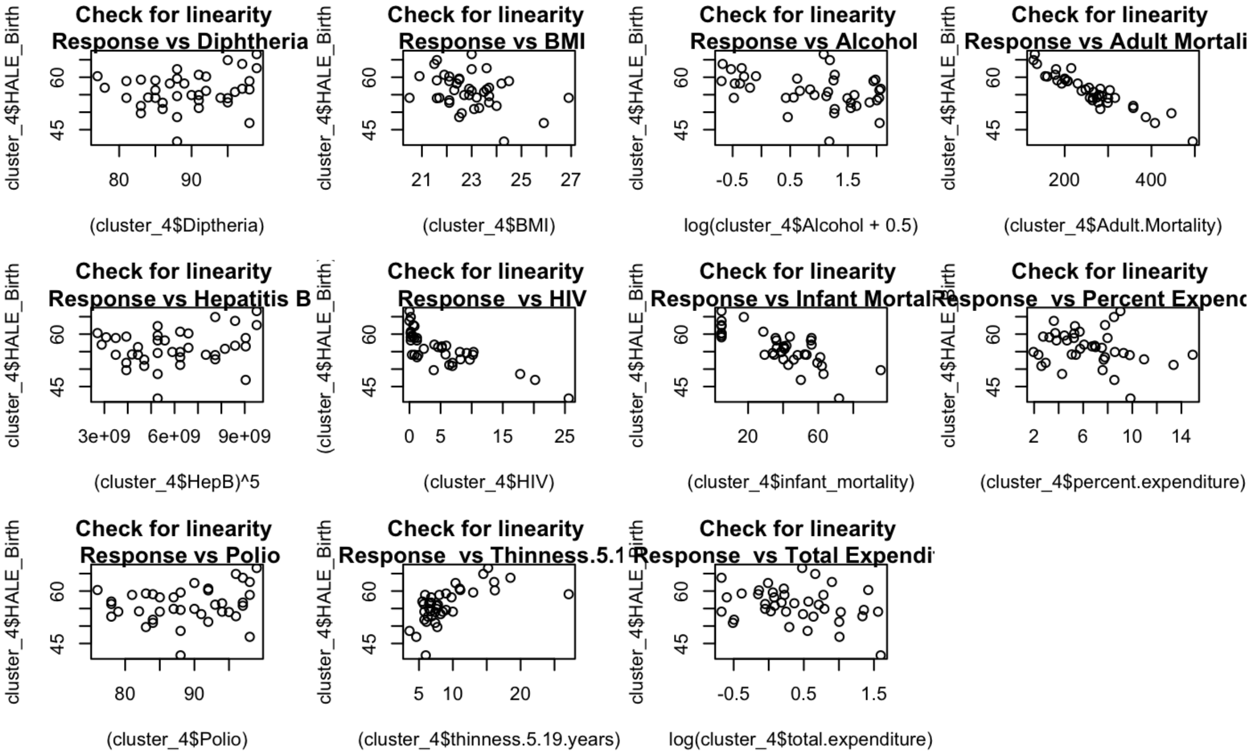
Appendix O: Residuals vs fitted values plot (cluster 3)



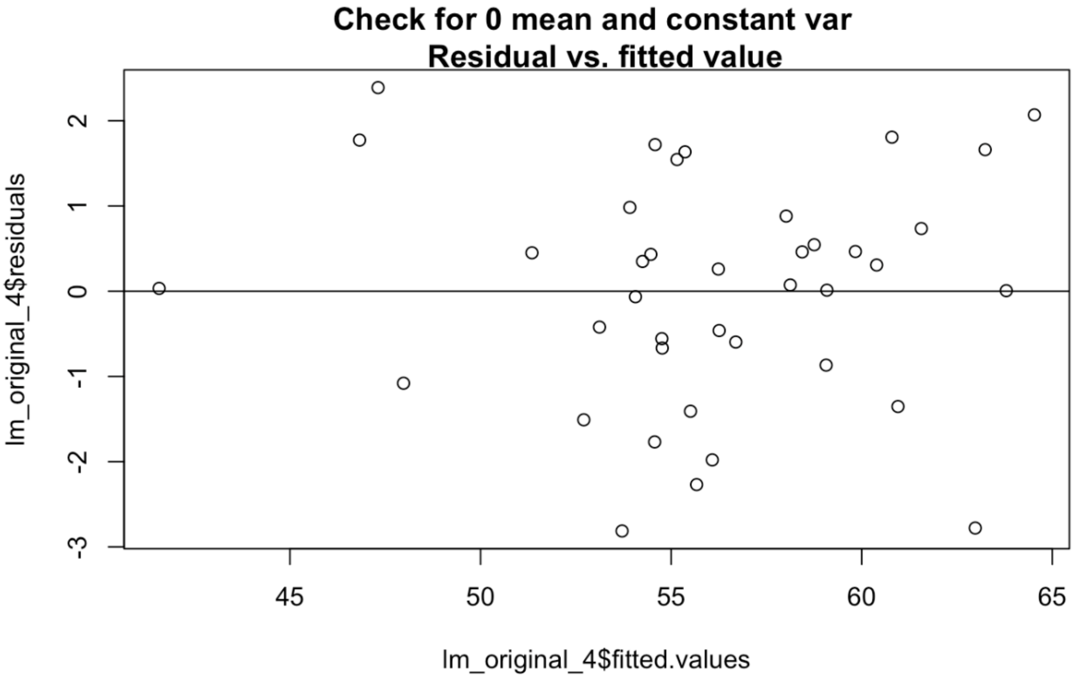
Appendix P: Linearity plots for regressor variables before transformation (cluster 4)



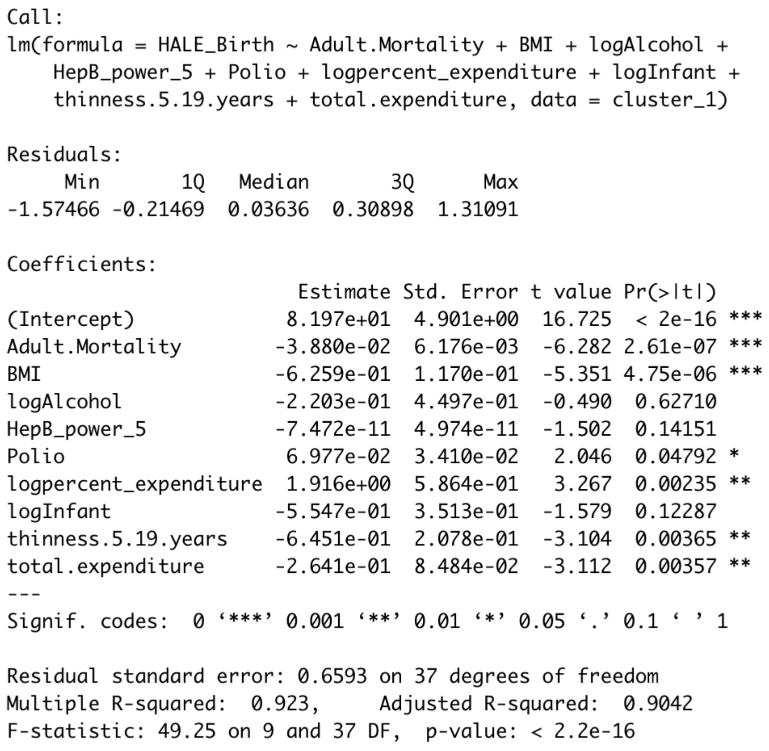
Appendix Q: Linearity plots for regressor variables after transformation (cluster 4)



Appendix R: Residuals vs fitted values plot (cluster 4)



Appendix S: Final Model for cluster 1



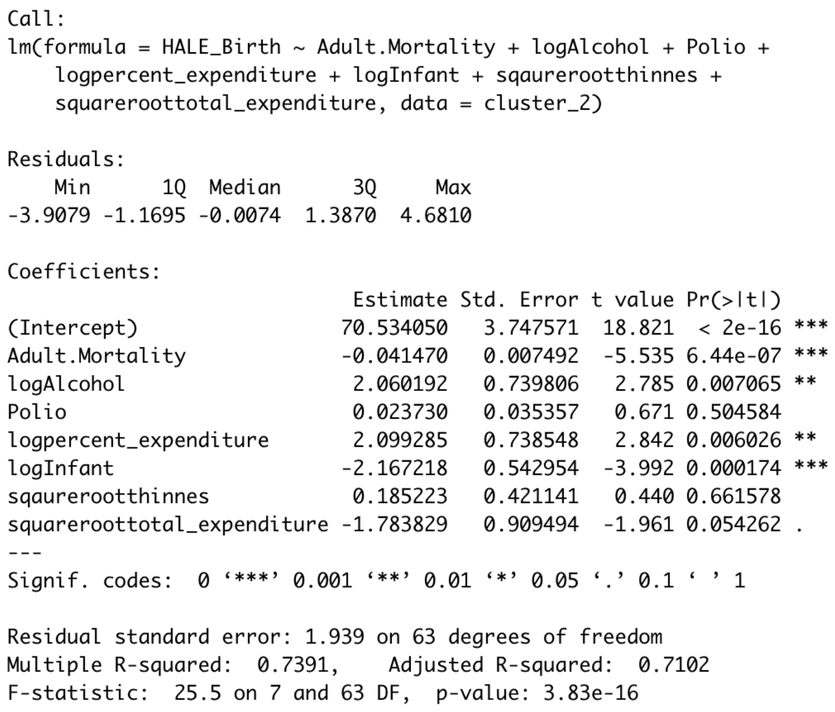
```
Call:
lm(formula = HALE_Birth ~ Adult.Mortality + BMI + logAlcohol + 
    HepB_power_5 + Polio + logpercent_expenditure + logInfant + 
    thinness.5.19.years + total.expenditure, data = cluster_1)

Residuals:
     Min       1Q   Median       3Q      Max 
-1.57466 -0.21469  0.03636  0.30898  1.31091 

Coefficients:
                        Estimate Std. Error t value Pr(>|t|)    
(Intercept)            8.197e+01  4.901e+00  16.725  < 2e-16 ***
Adult.Mortality       -3.880e-02  6.176e-03  -6.282 2.61e-07 ***
BMI                   -6.259e-01  1.170e-01  -5.351 4.75e-06 ***
logAlcohol            -2.203e-01  4.497e-01  -0.490  0.62710    
HepB_power_5          -7.472e-11  4.974e-11  -1.502  0.14151    
Polio                  6.977e-02  3.410e-02   2.046  0.04792 *  
logpercent_expenditure  1.916e+00  5.864e-01   3.267  0.00235 ** 
logInfant             -5.547e-01  3.513e-01  -1.579  0.12287    
thinness.5.19.years   -6.451e-01  2.078e-01  -3.104  0.00365 ** 
total.expenditure     -2.641e-01  8.484e-02  -3.112  0.00357 ** 
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.6593 on 37 degrees of freedom
Multiple R-squared:  0.923,	Adjusted R-squared:  0.9042 
F-statistic: 49.25 on 9 and 37 DF,  p-value: < 2.2e-16
```

Appendix T: Final Model for cluster 2



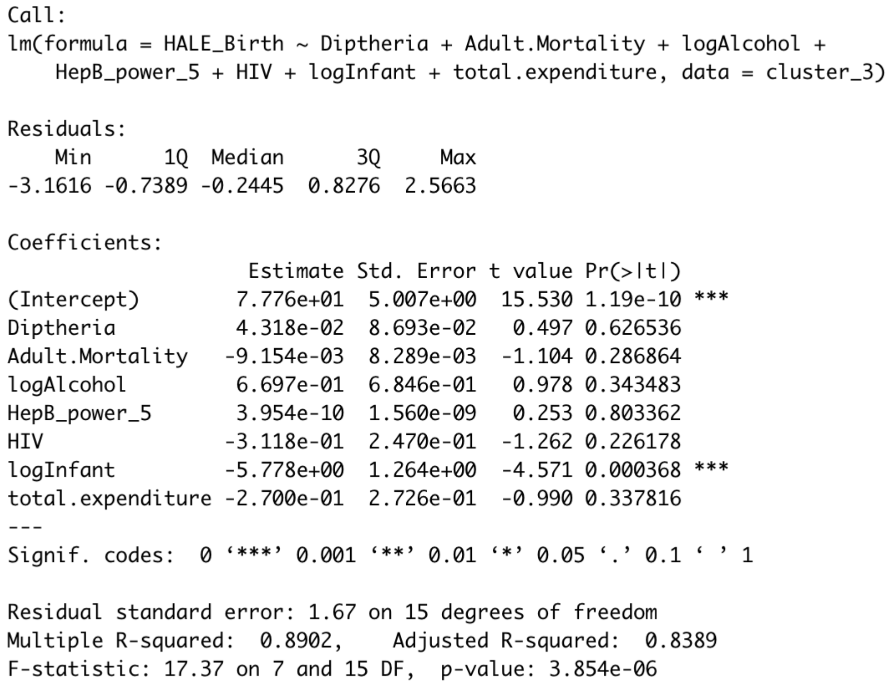
```
Call:
lm(formula = HALE_Birth ~ Adult.Mortality + logAlcohol + Polio + 
    logpercent_expenditure + logInfant + sqaurerootthinnes + 
    squareroottotal_expenditure, data = cluster_2)

Residuals:
    Min      1Q  Median      3Q     Max 
-3.9079 -1.1695 -0.0074  1.3870  4.6810 

Coefficients:
                             Estimate Std. Error t value Pr(>|t|)    
(Intercept)                 70.534050   3.747571  18.821  < 2e-16 ***
Adult.Mortality             -0.041470   0.007492  -5.535 6.44e-07 ***
logAlcohol                   2.060192   0.739806   2.785 0.007065 ** 
Polio                        0.023730   0.035357   0.671 0.504584    
logpercent_expenditure       2.099285   0.738548   2.842 0.006026 ** 
logInfant                   -2.167218   0.542954  -3.992 0.000174 ***
sqaurerootthinnes            0.185223   0.421141   0.440 0.661578    
squareroottotal_expenditure -1.783829   0.909494  -1.961 0.054262 .  
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.939 on 63 degrees of freedom
Multiple R-squared:  0.7391,	Adjusted R-squared:  0.7102 
F-statistic:  25.5 on 7 and 63 DF,  p-value: 3.83e-16
```

Appendix U: Final Model for cluster 3



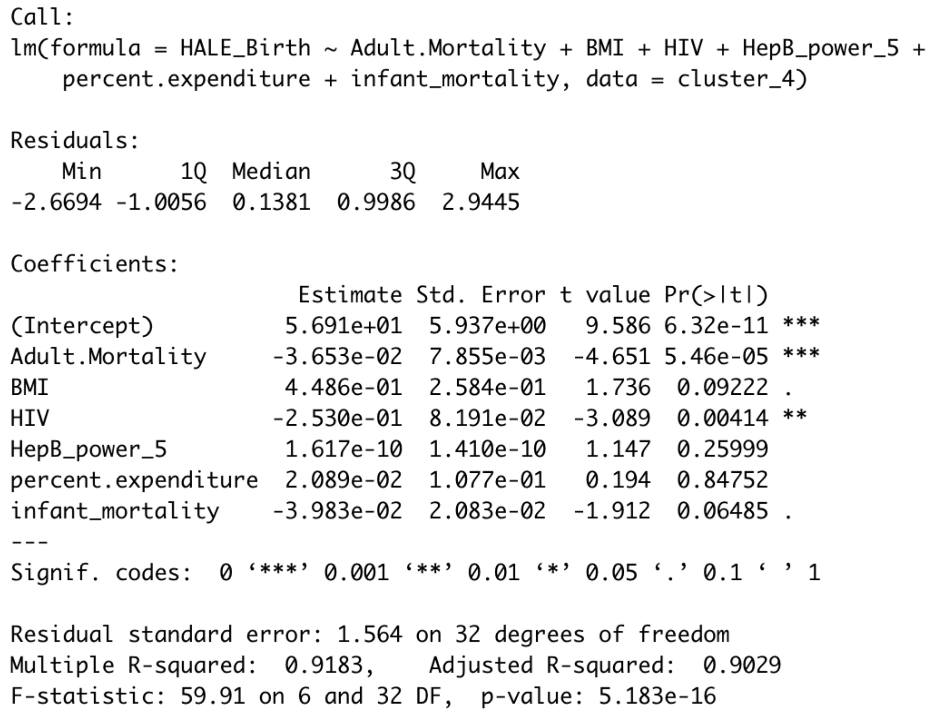
```
Call:
lm(formula = HALE_Birth ~ Diptheria + Adult.Mortality + logAlcohol + 
    HepB_power_5 + HIV + logInfant + total.expenditure, data = cluster_3)

Residuals:
    Min      1Q  Median      3Q     Max 
-3.1616 -0.7389 -0.2445  0.8276  2.5663 

Coefficients:
                    Estimate Std. Error t value Pr(>|t|)    
(Intercept)        7.776e+01  5.007e+00  15.530 1.19e-10 ***
Diptheria          4.318e-02  8.693e-02   0.497 0.626536    
Adult.Mortality   -9.154e-03  8.289e-03  -1.104 0.286864    
logAlcohol         6.697e-01  6.846e-01   0.978 0.343483    
HepB_power_5       3.954e-10  1.560e-09   0.253 0.803362    
HIV               -3.118e-01  2.470e-01  -1.262 0.226178    
logInfant         -5.778e+00  1.264e+00  -4.571 0.000368 ***
total.expenditure -2.700e-01  2.726e-01  -0.990 0.337816    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.67 on 15 degrees of freedom
Multiple R-squared:  0.8902,	Adjusted R-squared:  0.8389 
F-statistic: 17.37 on 7 and 15 DF,  p-value: 3.854e-06
```

Appendix V: Final Model for cluster 4



```
Call:
lm(formula = HALE_Birth ~ Adult.Mortality + BMI + HIV + HepB_power_5 + 
    percent.expenditure + infant_mortality, data = cluster_4)

Residuals:
    Min      1Q  Median      3Q     Max 
-2.6694 -1.0056  0.1381  0.9986  2.9445 

Coefficients:
                      Estimate Std. Error t value Pr(>|t|)    
(Intercept)          5.691e+01  5.937e+00   9.586 6.32e-11 ***
Adult.Mortality     -3.653e-02  7.855e-03  -4.651 5.46e-05 ***
BMI                  4.486e-01  2.584e-01   1.736  0.09222 .  
HIV                 -2.530e-01  8.191e-02  -3.089  0.00414 ** 
HepB_power_5         1.617e-10  1.410e-10   1.147  0.25999    
percent.expenditure  2.089e-02  1.077e-01   0.194  0.84752    
infant_mortality    -3.983e-02  2.083e-02  -1.912  0.06485 .  
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.564 on 32 degrees of freedom
Multiple R-squared:  0.9183,	Adjusted R-squared:  0.9029 
F-statistic: 59.91 on 6 and 32 DF,  p-value: 5.183e-16
```