

# STAT 420 Final Data Analysis Project - Team KWS

## Introduction

Life expectancy varies greatly across the world. Understanding the factors that influence life expectancy can provide insights for policymakers, healthcare providers, and individuals. This project aims to delve into the determinants of life expectancy.

The dataset used is from the World Health Organization (WHO) and United Nations, which provides a wealth of data related to life expectancy across 193 countries. The dataset, publicly available on Kaggle under the title "Life Expectancy (WHO)", comprises 22 columns and 2,938 observations. Each observation represents the state of a given country in a particular year, with 20 predictor variables providing comprehensive information about various factors potentially impacting life expectancy.

The predictor variables span across a broad spectrum, including immunization-related factors, mortality factors, economic indicators, and social factors. Among these, we believe certain variables like the status of the country (developed vs developing), infant mortality rate, adult mortality rate, health expenditure, GDP, and average years of schooling may have significant importance in predicting life expectancy.

Our objective in this project is to create a model that can effectively predict life expectancy based on these predictors. We're particularly interested in this dataset due to our curiosity about the factors influencing life expectancy and our desire to uncover the differences between countries that lead to a wide range of life expectancies. By building this model, we aim to provide insights that could potentially guide health policy and contribute to improvements in global health outcomes.

## Methods

### Loading Data

```
#read data
library(readr)
life_expectancy_data_full = read_csv("Life Expectancy Data.csv")
```

```
## Rows: 2938 Columns: 22
## — Column specification —————
## Delimiter: ","
## chr  (2): Country, Status
## dbl (20): Year, Life expectancy, Adult Mortality, infant deaths, Alcohol, pe...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

We will remove rows with empty values to make analysis easier.

```
#remove rows with empty vals
life_expectancy_data = na.omit(life_expectancy_data_full)
#View(life_expectancy_data)
```

For our analysis, we have chosen to convert the predictors Status and Country into factor variables, so that we can treat them as categories.

```
#convert Status (levels: Developing, Developed) to factor variable
life_expectancy_data$Status = as.factor(life_expectancy_data$Status)

#convert Country to factor variable
life_expectancy_data$Country = as.factor(life_expectancy_data$Country)
```

## Exploratory Analysis

The pairs plot of all of the variables was very large, so we will display correlations between the numeric variables in the dataset with a Correlogram instead.

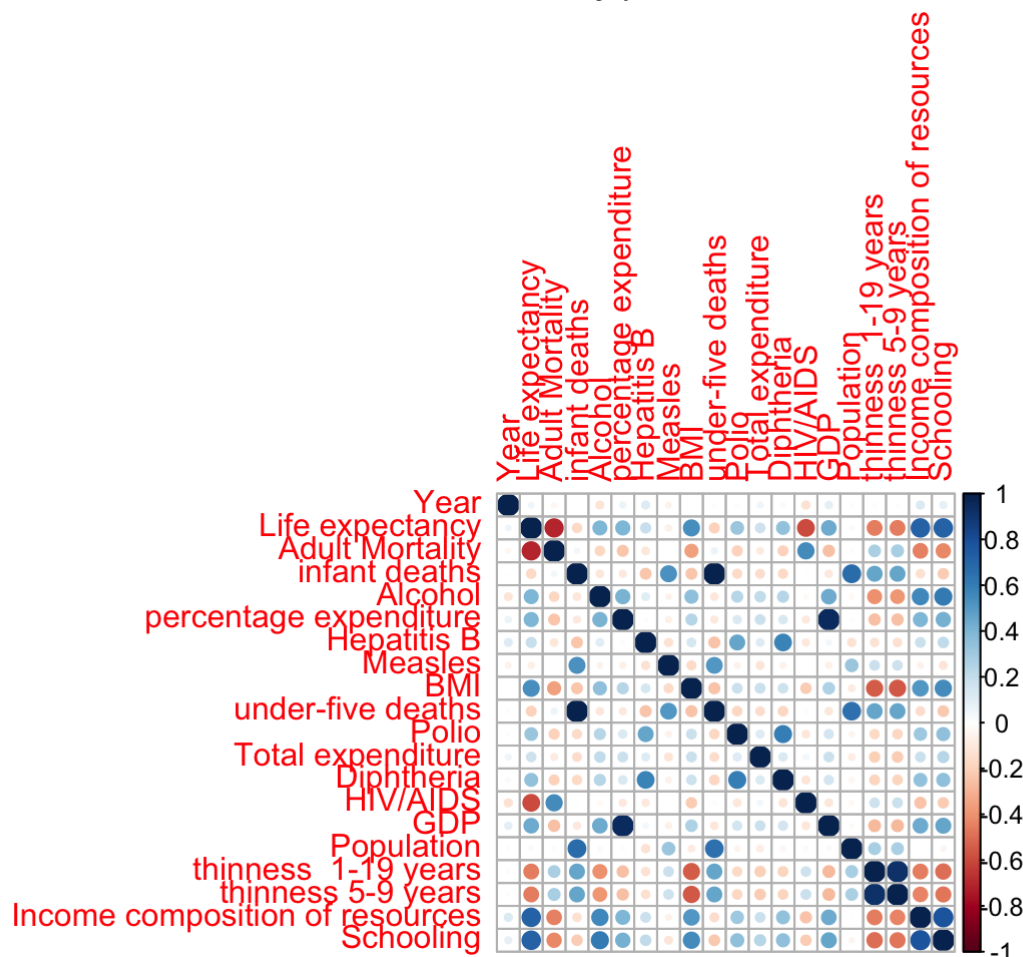
```
#pairs(life_expectancy_data)

#find pairwise correlations between all numeric variables in dataset
numeric_cols =  unlist(lapply(life_expectancy_data, is.numeric))
life_expectancy_data_numeric = life_expectancy_data[ , numeric_cols]
#View(life_expectancy_data_numeric)

correlations = cor(life_expectancy_data_numeric)

#plot correlogram (positive correlations are blue, negative correlations are red)
library(corrplot)

corrplot(correlations)
```



We notice that there are a few variables with very high correlations, so we will attempt to address this issue in our modeling.

## Initial Additive Model

First, we create a main effects model using all of the available predictors in the dataset to predict life expectancy.

```
additive_model = lm(`Life expectancy` ~ ., data = life_expectancy_data)
#summary(additive_model)
```

# Analyze Model

```
#Calculate LOOCV RMSE
calc_loocv_rmse = function(model) {
  sqrt(mean((resid(model) / (1 - hatvalues(model))) ^ 2))
}

# Function to calculate model metrics

calc_model_metrics = function(model) {
  # Calculate adjusted R squared
  model_adj_r2 = summary(model)$adj.r.squared

  # Calculate RMSE
  model_rmse = sqrt(mean(resid(model) ^ 2))

  # Calculate LOOCV RMSE
  model_loocv_rmse = sqrt(mean((resid(model) / (1 - hatvalues(model))) ^ 2))

  # Calculate AIC
  model_AIC = extractAIC(model)[2] # get only the AIC value, not the degrees of freedom

  # Return a list containing the calculated metrics
  list(Adj_R_Squared = model_adj_r2, RMSE = model_rmse, LOOCV_RMSE = model_loocv_rmse, AIC = model_AIC)
}

additive_model_metrics = as.data.frame(calc_model_metrics(additive_model))

library(knitr)
knitr::kable(additive_model_metrics, caption = "Additive Model Summary", digits = 4)
```

## Additive Model Summary

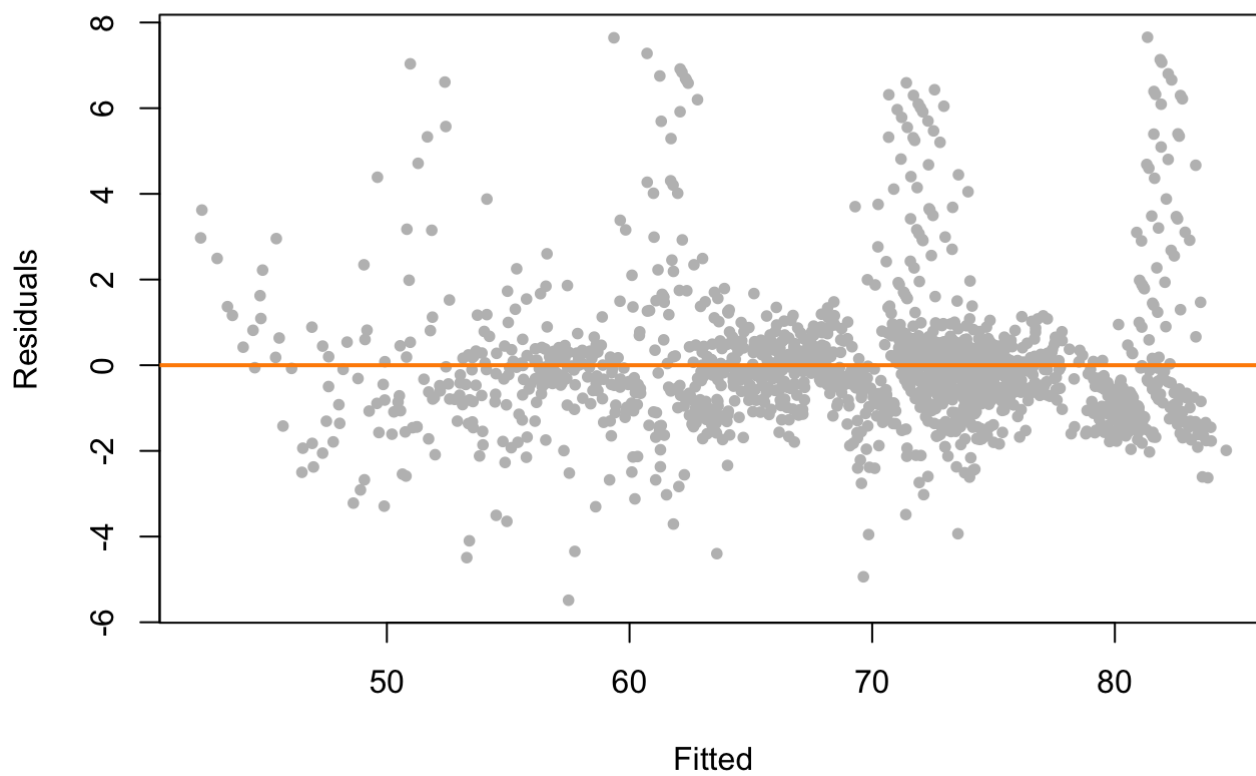
Adj_R_Squared	RMSE	LOOCV_RMSE	AIC
0.9642	1.5859	1.7559	1824.977

# Check model assumptions

```
#Fitted vs Residuals Plot

plot(fitted(additive_model), resid(additive_model), col = "grey", pch = 20,
      xlab = "Fitted", ylab = "Residuals", main = "Data from Model 1")
abline(h = 0, col = "darkorange", lwd = 2)
```

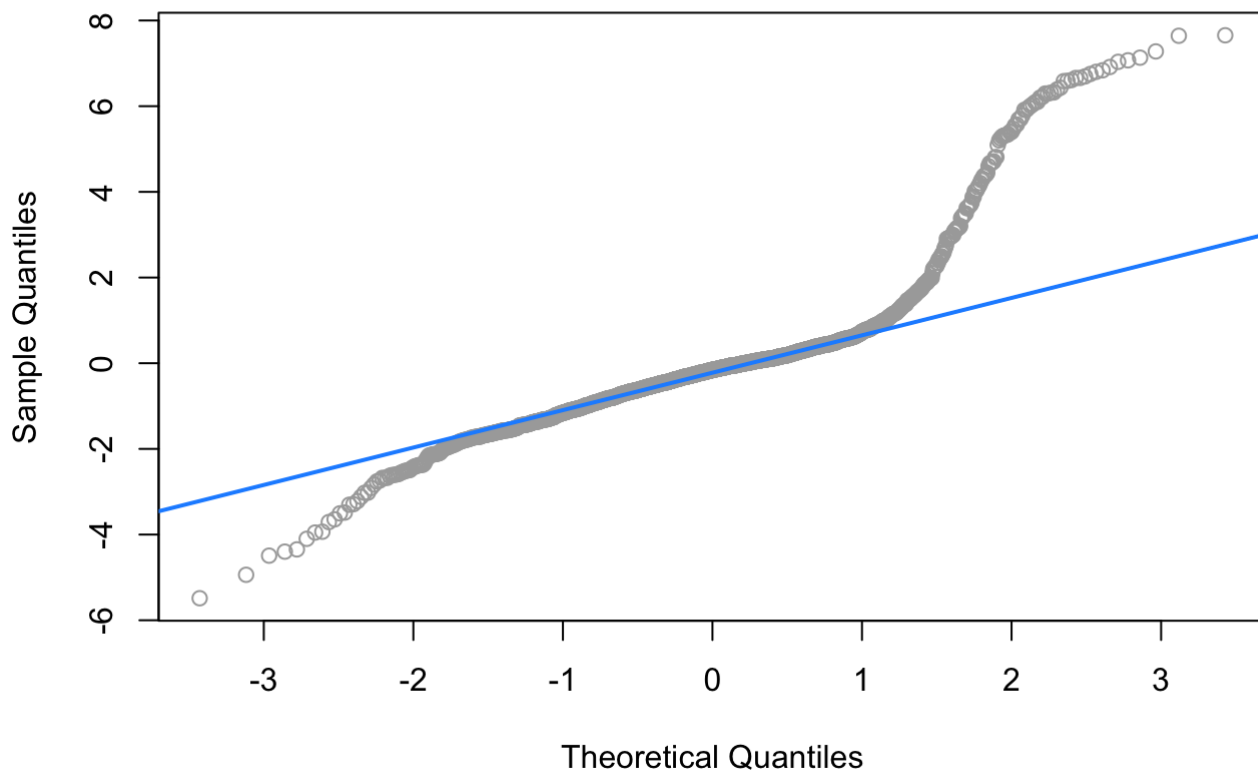
## Data from Model 1



*#Q-Q Plot*

```
qqnorm(resid(additive_model), main = "Normal Q-Q Plot, fit_1", col = "darkgrey")  
qqline(resid(additive_model), col = "dodgerblue", lwd = 2)
```

## Normal Q-Q Plot, fit\_1



The fitted vs residuals plot and Q-Q plot suggest issues with this model's assumptions, which we will confirm via more formal tests.

```
library(lmtest)
bptest(additive_model)
```

```
##
## studentized Breusch-Pagan test
##
## data: additive_model
## BP = 283.35, df = 151, p-value = 4.012e-10
```

```
shapiro.test(resid(additive_model))
```

```
##
## Shapiro-Wilk normality test
##
## data: resid(additive_model)
## W = 0.82427, p-value < 2.2e-16
```

The results of the Breusch-Pagan test and the Shapiro-Wilk test confirm that this model violates both the constant variance assumption and the normality assumption. We will attempt to improve these issues below.

## Model Improvement: Removing Highly Correlated Variables

We will attempt to fix these issues by removing highly correlated predictors. In particular, the variables `thinness 1-19 years` and `thinness 5-9 years` had a correlation of 0.997. In addition, the variables `under 5 death` and `infant death` had a correlation of 0.928. We'll remove one from each of the above pairs of predictors and reassess the model.

```

#----- Next step -----

#Removing infant deaths and thinness 1-19 years
corr_removed_data_1 = life_expectancy_data[ , !(names(life_expectancy_data) %in% c('infant deaths', 'thinness 1-19 years'))]
corr_removed_model_1 = lm(`Life expectancy` ~ ., data = corr_removed_data_1)
corr_removed_metrics_1 = calc_model_metrics(corr_removed_model_1)

#Removing infant deaths and thinness 5-9 years
corr_removed_data_2 = life_expectancy_data[ , !(names(life_expectancy_data) %in% c('infant deaths', 'thinness 5-9 years'))]
corr_removed_model_2 = lm(`Life expectancy` ~ ., data = corr_removed_data_2)
corr_removed_metrics_2 = calc_model_metrics(corr_removed_model_2)

#Removing under 5 deaths and thinness 1-19 years
corr_removed_data_3 = life_expectancy_data[ , !(names(life_expectancy_data) %in% c('under 5 death', 'thinness 1-19 years'))]
corr_removed_model_3 = lm(`Life expectancy` ~ ., data = corr_removed_data_3)
corr_removed_metrics_3 = calc_model_metrics(corr_removed_model_3)

#Removing under 5 deaths and thinness 5-9 years
corr_removed_data_4 = life_expectancy_data[ , !(names(life_expectancy_data) %in% c('under 5 death', 'thinness 5-9 years'))]
corr_removed_model_4 = lm(`Life expectancy` ~ ., data = corr_removed_data_4)
corr_removed_metrics_4 = calc_model_metrics(corr_removed_model_4)

# Create a data frame summarizing the model metrics
model_summary = data.frame(
  Variables_Removed = c("infant deaths + thinness 1-19", "infant deaths + thinness 5-9",
    "under 5 deaths + thinness 1-19", "under 5 deaths + thinness 5-9 years"),
  Adjusted_R_Squared = c(corr_removed_metrics_1$Adj_R_Squared, corr_removed_metrics_2$Adj_R_Squared,
    corr_removed_metrics_3$Adj_R_Squared, corr_removed_metrics_4$Adj_R_Squared),
  RMSE = c(corr_removed_metrics_1$RMSE, corr_removed_metrics_2$RMSE, corr_removed_metrics_3$RMSE,
    corr_removed_metrics_4$RMSE),
  LOOCV_RMSE = c(corr_removed_metrics_1$LOOCV_RMSE, corr_removed_metrics_2$LOOCV_RMSE,
    corr_removed_metrics_3$LOOCV_RMSE, corr_removed_metrics_4$LOOCV_RMSE),
  AIC = c(corr_removed_metrics_1$AIC, corr_removed_metrics_2$AIC, corr_removed_metrics_3$AIC,
    corr_removed_metrics_4$AIC)
)

library(knitr)
knitr::kable(model_summary, caption = "Model Summary", digits = 4)

```

## Model Summary

Variables_Removed	Adjusted_R_Squared	RMSE	LOOCV_RMSE	AIC
infant deaths + thinness 1-19	0.9640	1.5914	1.7576	1834.247



Variables_Removed	Adjusted_R_Squared	RMSE	LOOCV_RMSE	AIC
infant deaths + thinness 5-9	0.9639	1.5937	1.7590	1837.088
under 5 deaths + thinness 1-19	0.9642	1.5859	1.7559	1824.977
under 5 deaths + thinness 5-9 years	0.9641	1.5883	1.7572	1827.939

Our analysis suggests that removing the variables `under 5 deaths` and `thinness 1-19` results in the most model improvement. However, the resulting AIC of this model is still large, so we need to further improve the model.

## Model Improvement: Log Transformation of Predictor Variables

We will attempt to further improve the model by log transforming certain predictor variables (`Population` and `GDP`).

```
#----- Next step -----

log_transform_1 = lm(`Life expectancy` ~ . + log(Population) + log(GDP), data = life_exp
ectancy_data)
log_transformed_metrics_1 = calc_model_metrics(log_transform_1)

log_transform_2 = lm(`Life expectancy` ~ . + log(GDP), data = life_expectancy_data)
log_transformed_metrics_2 = calc_model_metrics(log_transform_2)

log_transform_3 = lm(`Life expectancy` ~ . + log(Population), data = life_expectancy_dat
a)
log_transformed_metrics_3 = calc_model_metrics(log_transform_3)

log_model_summary = data.frame(
  Variables_Log_Transformed = c("Population + GDP", "GDP", "Population"),
  Adjusted_R_Squared = c(log_transformed_metrics_1$Adj_R_Squared, log_transformed_metric
s_2$Adj_R_Squared, log_transformed_metrics_3$Adj_R_Squared),
  RMSE = c(log_transformed_metrics_1$RMSE, log_transformed_metrics_2$RMSE, log_transform
ed_metrics_3$RMSE),
  LOOCV_RMSE = c(log_transformed_metrics_1$LOOCV_RMSE, log_transformed_metrics_2$LOOCV_R
MSE, log_transformed_metrics_3$LOOCV_RMSE),
  AIC = c(log_transformed_metrics_1$AIC, log_transformed_metrics_2$AIC, log_transformed_
metrics_3$AIC)
)

knitr::kable(log_model_summary, caption = "Log Transformed Model Summary", digits = 4)
```

### Log Transformed Model Summary

Variables_Log_Transformed	Adjusted_R_Squared	RMSE	LOOCV_RMSE	AIC
Population + GDP	0.9642	1.5853	1.7583	1827.612
GDP	0.9642	1.5853	1.7569	1825.613

Variables_Log_Transformed	Adjusted_R_Squared	RMSE	LOOCV_RMSE	AIC
Population	0.9642	1.5859	1.7574	1826.977

Experimenting with various log transformations of the predictor variables `Population` and `GDP` suggests that log transformation of `GDP` resulted in a minimal improvement in AIC.

The resulting AIC for this model is still rather large, so we will use AIC and BIC backwards search to attempt to further improve the model.

## Model Improvement: Quality Criterion & Backwards Search Procedure

First, we consider both AIC and BIC on the initial additive model containing all predictor variables.

```
#----- Next step -----

n = length(resid(additive_model))

#AIC and BIC for additive models
additive_mod_back_aic = step(additive_model, direction = "backward", trace = 0)
additive_mod_back_bic = step(additive_model, direction = "backward", k = log(n), trace = 0)

metrics_additive_mod_back_aic = calc_model_metrics(additive_mod_back_aic)
metrics_additive_mod_back_bic = calc_model_metrics(additive_mod_back_bic)
```

```
additive_model_summary = data.frame(
  Criterion = c("additive AIC", "Additive BIC"),

  Adjusted_R_Squared =
    c(metrics_additive_mod_back_aic$Adj_R_Squared, metrics_additive_mod_back_bic$Adj_R_Squared),

  RMSE = c(metrics_additive_mod_back_aic$RMSE, metrics_additive_mod_back_bic$RMSE),

  LOOCV_RMSE =
    c(metrics_additive_mod_back_aic$LOOCV_RMSE, metrics_additive_mod_back_bic$LOOCV_RMSE),

  AIC = c(metrics_additive_mod_back_aic$AIC, metrics_additive_mod_back_bic$AIC)
)

knitr::kable(additive_model_summary, caption = "Additive model AIC and BIC Summary", digits = 4)
```

Additive model AIC and BIC Summary

Criterion	Adjusted_R_Squared	RMSE	LOOCV_RMSE	AIC
-----------	--------------------	------	------------	-----

Criterion	Adjusted_R_Squared	RMSE	LOOCV_RMSE	AIC
additive AIC	0.9643	1.5882	1.7430	1809.740
Additive BIC	0.9642	1.5932	1.7455	1814.022

We see that using AIC and a backwards search on the additive model resulted in a significant improvement in AIC, with an AIC of 1809.740.

Next, we consider using AIC and BIC on the models with highly correlated variables removed.

```
# AIC and BIC for models without highly correlated features
corr_mod_1_back_aic = step(corr_removed_model_1, direction = "backward", trace = 0)
corr_mod_2_back_aic = step(corr_removed_model_2, direction = "backward", trace = 0)
corr_mod_3_back_aic = step(corr_removed_model_3, direction = "backward", trace = 0)

corr_mod_1_back_bic = step(corr_removed_model_1, direction = "backward", k = log(n), trace = 0)
corr_mod_2_back_bic = step(corr_removed_model_2, direction = "backward", k = log(n), trace = 0)
corr_mod_3_back_bic = step(corr_removed_model_3, direction = "backward", k = log(n), trace = 0)

metrics_corr_mod_1_back_aic = calc_model_metrics(corr_mod_1_back_aic)
metrics_corr_mod_2_back_aic = calc_model_metrics(corr_mod_2_back_aic)
metrics_corr_mod_3_back_aic = calc_model_metrics(corr_mod_3_back_aic)

metrics_corr_mod_1_back_bic = calc_model_metrics(corr_mod_1_back_bic)
metrics_corr_mod_2_back_bic = calc_model_metrics(corr_mod_2_back_bic)
metrics_corr_mod_3_back_bic = calc_model_metrics(corr_mod_3_back_bic)
```

```

corr_model_summary = data.frame(
  Criterion = c("corr AIC model 1", "corr AIC model 2", "corr AIC model 3",
               "corr BIC model 1", "corr BIC model 2", "corr BIC model 3"
),

  Adjusted_R_Squared =
  c(metrics_corr_mod_1_back_aic$Adj_R_Squared, metrics_corr_mod_2_back_aic$Adj_R_Square
d,
    metrics_corr_mod_3_back_aic$Adj_R_Squared,
    metrics_corr_mod_1_back_bic$Adj_R_Squared, metrics_corr_mod_2_back_bic$Adj_R_Square
d,
    metrics_corr_mod_3_back_bic$Adj_R_Squared
  ),

  RMSE =
  c(metrics_corr_mod_1_back_aic$RMSE, metrics_corr_mod_2_back_aic$RMSE,
    metrics_corr_mod_3_back_aic$RMSE,
    metrics_corr_mod_1_back_bic$RMSE, metrics_corr_mod_2_back_bic$RMSE,
    metrics_corr_mod_3_back_bic$RMSE
  ),

  LOOCV_RMSE =
  c(metrics_corr_mod_1_back_aic$LOOCV_RMSE, metrics_corr_mod_2_back_aic$LOOCV_RMSE,
    metrics_corr_mod_3_back_aic$LOOCV_RMSE,
    metrics_corr_mod_1_back_bic$LOOCV_RMSE, metrics_corr_mod_2_back_bic$LOOCV_RMSE,
    metrics_corr_mod_3_back_bic$LOOCV_RMSE
  ),

  AIC = c(metrics_corr_mod_1_back_aic$AIC, metrics_corr_mod_2_back_aic$AIC,
    metrics_corr_mod_3_back_aic$AIC,
    metrics_corr_mod_1_back_bic$AIC, metrics_corr_mod_2_back_bic$AIC,
    metrics_corr_mod_3_back_bic$AIC
  )
)

knitr::kable(corr_model_summary, caption = "correlations removed model AIC and BIC Summa
ry", digits = 4)

```

correlations removed model AIC and BIC Summary

Criterion	Adjusted_R_Squared	RMSE	LOOCV_RMSE	AIC
corr AIC model 1	0.9641	1.5935	1.7457	1818.552
corr AIC model 2	0.9640	1.5958	1.7484	1823.343
corr AIC model 3	0.9643	1.5882	1.7430	1809.740
corr BIC model 1	0.9639	1.6003	1.7497	1824.704
corr BIC model 2	0.9638	1.6043	1.7531	1830.836
corr BIC model 3	0.9642	1.5932	1.7455	1814.022

The results show that using AIC and BIC on the additive model with highly correlated variables removed results in varying improvement in AIC.

We will investigate the effect of using these quality criteria and search procedures on the models with log transformations of certain predictors.

```
# AIC and BIC for models with log transformations
transform_mod_1_back_aic = step(log_transform_1, direction = "backward", trace = 0)
transform_mod_2_back_aic = step(log_transform_2, direction = "backward", trace = 0)
transform_mod_3_back_aic = step(log_transform_3, direction = "backward", trace = 0)

transform_mod_1_back_bic = step(log_transform_1, direction = "backward", k = log(n), trace = 0)
transform_mod_2_back_bic = step(log_transform_2, direction = "backward", k = log(n), trace = 0)
transform_mod_3_back_bic = step(log_transform_3, direction = "backward", k = log(n), trace = 0)

metrics_transform_mod_1_back_aic = calc_model_metrics(transform_mod_1_back_aic)
metrics_transform_mod_2_back_aic = calc_model_metrics(transform_mod_2_back_aic)
metrics_transform_mod_3_back_aic = calc_model_metrics(transform_mod_3_back_aic)

metrics_transform_mod_1_back_bic = calc_model_metrics(transform_mod_1_back_bic)
metrics_transform_mod_2_back_bic = calc_model_metrics(transform_mod_2_back_bic)
metrics_transform_mod_3_back_bic = calc_model_metrics(transform_mod_3_back_bic)
```

```

transform_model_summary = data.frame(
  Variables_Log_Transformed = c("Log transform AIC model 1", "Log transform AIC model
2", "Log transform AIC model 3",
                                "Log transform BIC model 1", "Log transformBIC model 2",
"Log transform BIC model 3" ),

  Adjusted_R_Squared =
    c(metrics_tranform_mod_1_back_aic$Adj_R_Squared, metrics_tranform_mod_2_back_aic$Adj_R
_Squared, metrics_tranform_mod_3_back_aic$Adj_R_Squared,

      metrics_tranform_mod_1_back_bic$Adj_R_Squared, metrics_tranform_mod_2_back_bic$Adj_R
_Squared, metrics_tranform_mod_3_back_bic$Adj_R_Squared
    ),

  RMSE =
    c(metrics_tranform_mod_1_back_aic$RMSE, metrics_tranform_mod_2_back_aic$RMSE,
      metrics_tranform_mod_3_back_aic$RMSE,
      metrics_tranform_mod_1_back_bic$RMSE, metrics_tranform_mod_2_back_bic$RMSE,
      metrics_tranform_mod_3_back_bic$RMSE
    ),

  LOOCV_RMSE =
    c(metrics_tranform_mod_1_back_aic$LOOCV_RMSE, metrics_tranform_mod_2_back_aic$LOOCV_
RMSE,
      metrics_tranform_mod_3_back_aic$LOOCV_RMSE,
      metrics_tranform_mod_1_back_bic$LOOCV_RMSE, metrics_tranform_mod_2_back_bic$LOOCV_RM
SE,
      metrics_tranform_mod_3_back_bic$LOOCV_RMSE
    ),

  AIC = c(metrics_tranform_mod_1_back_aic$AIC, metrics_tranform_mod_2_back_aic$AIC,
      metrics_tranform_mod_3_back_aic$AIC,
      metrics_tranform_mod_1_back_bic$AIC, metrics_tranform_mod_2_back_bic$AIC,
      metrics_tranform_mod_3_back_bic$AIC
    )
)

knitr::kable(transform_model_summary, caption = "log transformation model AIC and BIC Su
mmmary", digits = 4)

```

#### log transformation model AIC and BIC Summary

Variables_Log_Transformed	Adjusted_R_Squared	RMSE	LOOCV_RMSE	AIC
Log transform AIC model 1	0.9643	1.5882	1.7430	1809.740
Log transform AIC model 2	0.9643	1.5882	1.7430	1809.740
Log transform AIC model 3	0.9643	1.5882	1.7430	1809.740
Log transform BIC model 1	0.9642	1.5932	1.7455	1814.022
Log transformBIC model 2	0.9642	1.5932	1.7455	1814.022

Variables_Log_Transformed	Adjusted_R_Squared	RMSE	LOOCV_RMSE	AIC
Log transform BIC model 3	0.9642	1.5932	1.7455	1814.022

Interestingly, we see that using AIC and backwards search on each of the models with log transformations of the predictors resulted in an improved AIC of 1809.74. This was the maximum improvement observed during our analysis.

## Results

Our analysis suggests that using AIC and a backwards search on the model with the highly correlated variables (under 5 deaths + thinness 1-19) removed resulted in the maximum improvement in AIC, which equaled the improvement in AIC found by using AIC and a backwards search on either of the models containing a log transformation of the predictors. These models both resulted in an AIC of 1809.740 and a LOOCV-RMSE of 1.7430. Their adjusted  $R^2$  equaled 0.9643, suggesting high goodness-of-fit for both of these models.

But can we trust these models' results? Let's examine each model's assumptions to see which one is more reliable. First we will examine the model with the correlated variables removed.

```
corr_model3_summary = data.frame(
  Criterion = c("corr AIC model 3"),
  Adjusted_R_Squared = c(metrics_corr_mod_3_back_aic$Adj_R_Squared),
  RMSE = c(metrics_corr_mod_3_back_aic$RMSE),
  LOOCV_RMSE = c(metrics_corr_mod_3_back_aic$LOOCV_RMSE),
  AIC = c(metrics_corr_mod_3_back_aic$AIC)
)

knitr::kable(corr_model3_summary, caption = "correlations removed model AIC and BIC Summary", digits = 4)
```

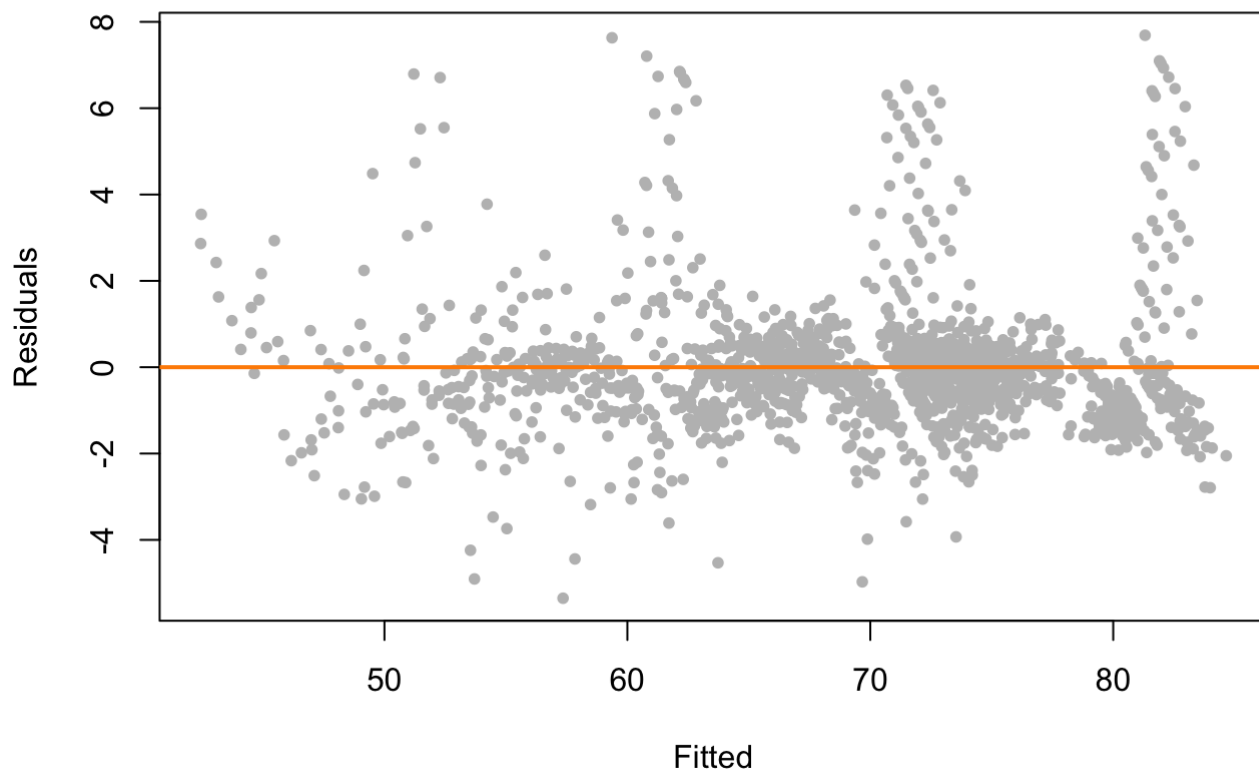
correlations removed model AIC and BIC Summary

Criterion	Adjusted_R_Squared	RMSE	LOOCV_RMSE	AIC
corr AIC model 3	0.9643	1.5882	1.743	1809.74

*#Fitted vs Residuals Plot*

```
plot(fitted(corr_mod_3_back_aic), resid(corr_mod_3_back_aic), col = "grey", pch = 20,
     xlab = "Fitted", ylab = "Residuals", main = "Data from Model 1")
abline(h = 0, col = "darkorange", lwd = 2)
```

## Data from Model 1

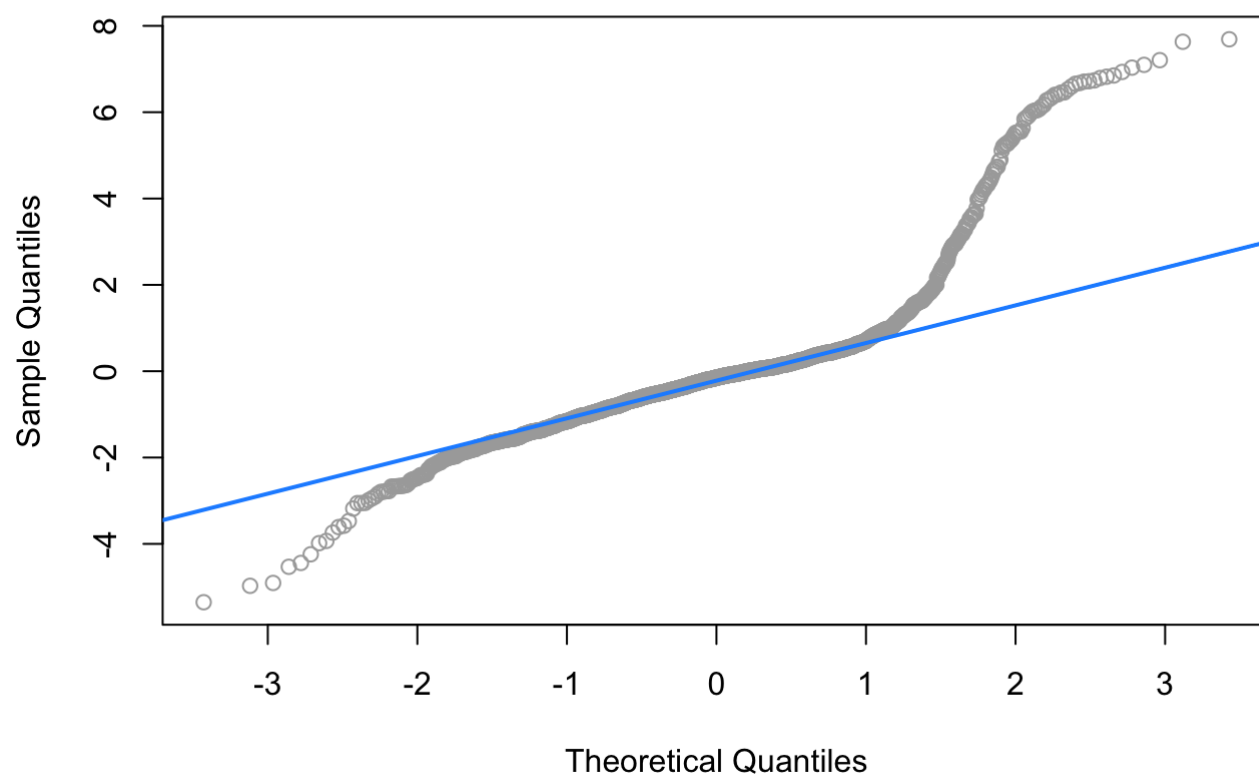


*#Q-Q Plot*

```
qqnorm(resid(corr_mod_3_back_aic), main = "Normal Q-Q Plot, fit_1", col = "darkgrey")  
qqline(resid(corr_mod_3_back_aic), col = "dodgerblue", lwd = 2)
```



## Normal Q-Q Plot, fit\_1



```
bptest(corr_mod_3_back_aic)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  corr_mod_3_back_aic
## BP = 280.98, df = 141, p-value = 2.419e-11
```

```
shapiro.test(resid(corr_mod_3_back_aic))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  resid(corr_mod_3_back_aic)
## W = 0.82406, p-value < 2.2e-16
```

```
anova(corr_mod_3_back_aic, additive_model)
```

```
## Analysis of Variance Table
##
## Model 1: `Life expectancy` ~ Country + Year + `infant deaths` + Alcohol +
##   `Hepatitis B` + `under-five deaths` + `HIV/AIDS` + `thinness 5-9 years` +
##   `Income composition of resources` + Schooling
## Model 2: `Life expectancy` ~ Country + Year + Status + `Adult Mortality` +
##   `infant deaths` + Alcohol + `percentage expenditure` + `Hepatitis B` +
##   Measles + BMI + `under-five deaths` + Polio + `Total expenditure` +
##   Diphtheria + `HIV/AIDS` + GDP + Population + `thinness 1-19 years` +
##   `thinness 5-9 years` + `Income composition of resources` +
##   Schooling
##   Res.Df    RSS Df Sum of Sq    F Pr(>F)
## 1    1507 4159.6
## 2    1497 4147.6 10    11.997 0.433 0.931
```

The results suggest that this model still violates the constant variance and normality assumptions. So, although the predictive power of this model appears strong, we cannot trust the inferences that are made from this model. In spite of this, an ANOVA F-test suggests that this model is preferred over the larger additive model containing all predictors for predicting life expectancy.

Next we will examine our alternative model with the best improvement in AIC: the model created from using AIC and a backwards search on the log transformed model.

```
tranform_mod_1_back_aic_summary = data.frame(
  Variables_Log_Transformed = c("Log transform AIC model 1"),
  Adjusted_R_Squared = c(metrics_tranform_mod_1_back_aic$Adj_R_Squared),
  RMSE = c(metrics_tranform_mod_1_back_aic$RMSE),
  LOOCV_RMSE = c(metrics_tranform_mod_1_back_aic$LOOCV_RMSE),
  AIC = c(metrics_tranform_mod_1_back_aic$AIC)
)

knitr::kable(tranform_mod_1_back_aic_summary, caption = "log transformation model AIC and BIC Summary", digits = 4)
```

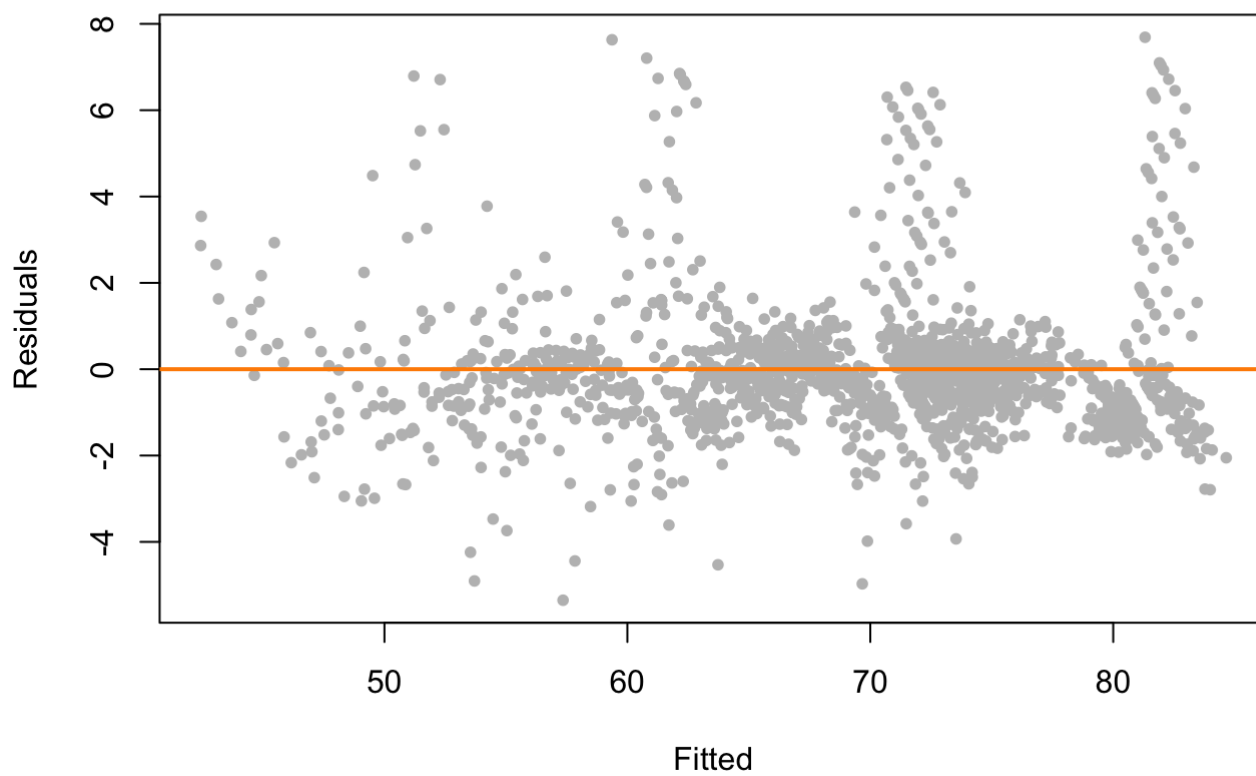
log transformation model AIC and BIC Summary

Variables_Log_Transformed	Adjusted_R_Squared	RMSE	LOOCV_RMSE	AIC
Log transform AIC model 1	0.9643	1.5882	1.743	1809.74

```
#Fitted vs Residuals Plot

plot(fitted(tranform_mod_1_back_aic), resid(tranform_mod_1_back_aic), col = "grey", pch
= 20,
      xlab = "Fitted", ylab = "Residuals", main = "Data from Model 1")
abline(h = 0, col = "darkorange", lwd = 2)
```

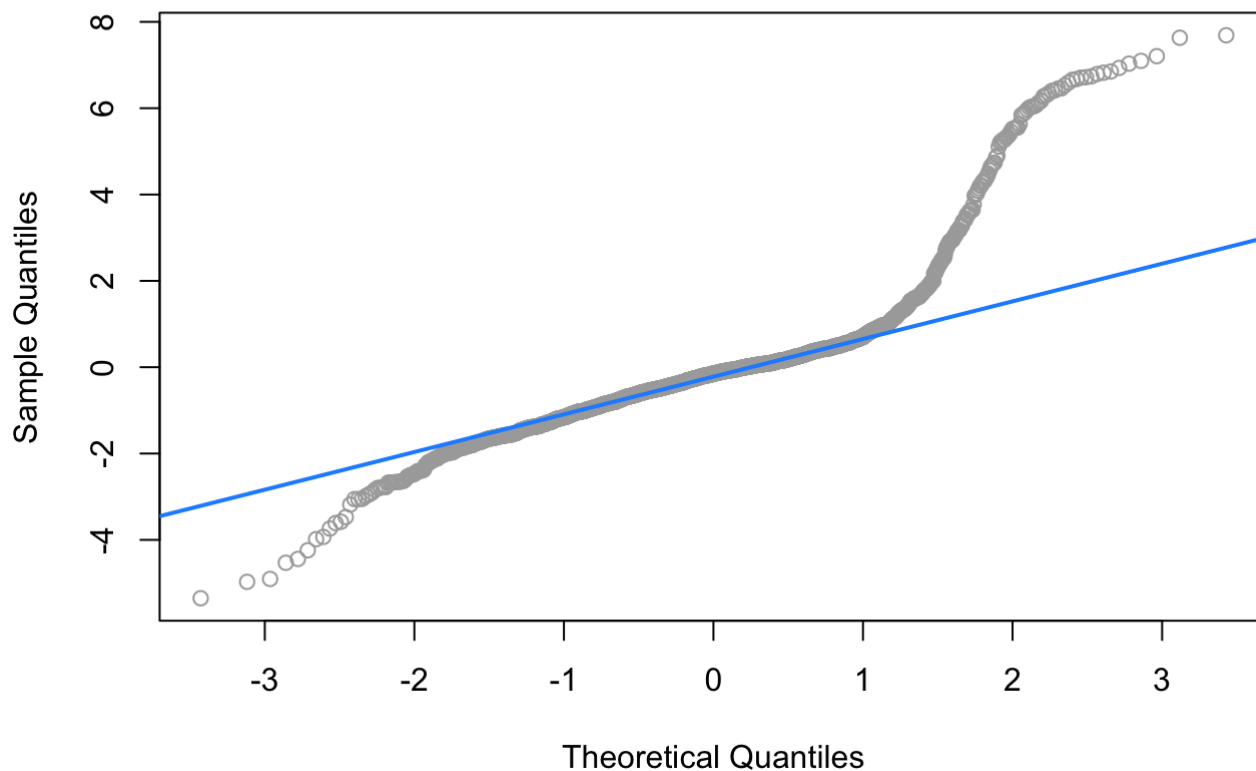
## Data from Model 1



*#Q-Q Plot*

```
qqnorm(resid(tranform_mod_1_back_aic), main = "Normal Q-Q Plot, fit_1", col = "darkgrey")  
qqline(resid(tranform_mod_1_back_aic), col = "dodgerblue", lwd = 2)
```

## Normal Q-Q Plot, fit\_1



```
bptest(tranform_mod_1_back_aic)
```

```
##
## studentized Breusch-Pagan test
##
## data: tranform_mod_1_back_aic
## BP = 280.98, df = 141, p-value = 2.419e-11
```

```
shapiro.test(resid(tranform_mod_1_back_aic))
```

```
##
## Shapiro-Wilk normality test
##
## data: resid(tranform_mod_1_back_aic)
## W = 0.82406, p-value < 2.2e-16
```

The results suggest that, despite the attempts at improvement, this model also violates the constant variance and normality assumptions.

```
anova(tranform_mod_1_back_aic, additive_model)
```

```
## Analysis of Variance Table
##
## Model 1: `Life expectancy` ~ Country + Year + `infant deaths` + Alcohol +
##   `Hepatitis B` + `under-five deaths` + `HIV/AIDS` + `thinness 5-9 years` +
##   `Income composition of resources` + Schooling
## Model 2: `Life expectancy` ~ Country + Year + Status + `Adult Mortality` +
##   `infant deaths` + Alcohol + `percentage expenditure` + `Hepatitis B` +
##   Measles + BMI + `under-five deaths` + Polio + `Total expenditure` +
##   Diphtheria + `HIV/AIDS` + GDP + Population + `thinness 1-19 years` +
##   `thinness 5-9 years` + `Income composition of resources` +
##   Schooling
##   Res.Df    RSS Df Sum of Sq    F Pr(>F)
## 1    1507 4159.6
## 2    1497 4147.6 10    11.997 0.433 0.931
```

Despite this, comparing this model with the full additive model suggests that we prefer this smaller model for predicting life expectancy.

## Discussions

In this section, we will delve into the implications and interpretation of the results, connecting the Introduction, Methods, and Results to draw meaningful conclusions.

When we picked this data set from the World health organization (WHO), our primary goal was to identify the key predictors that affect the life expectancy of the countries and use these predictors to come up with hypotheses and form models and methods based on these hypotheses and provide meaningful insights based on our discoveries.

Upon investigating the data set, we noticed some highly correlated variables, such as thinness 1-19 years and thinness 5-9 years, had a correlation of 0.997, and the variables under 5 death and infant death had a correlation of 0.928. We hypothesize that these variables created noise for our models to make accurate predictions. We have also applied log transformations to our data to improve the model performance. After building 6 different enhanced and transformed models, we decided to go with the simple additive model as our final model as we did not see any significant improvement in the performance of our model.

In conclusion, Our top models likely make acceptable suggestions for predicting life expectancy from predictors within the range of these predictor values. Still, their violation of linear model assumptions prevents us from making probabilistic statements like confidence intervals with these models. In other words, we might get reasonable predictions for the response variable using these models. Still, the estimates of the regression coefficients have increased variance and need to be more beneficial for inference.

## Appendix

Team Members: Saitejas Mopuri, Kimberly Martin, Woo Jeon