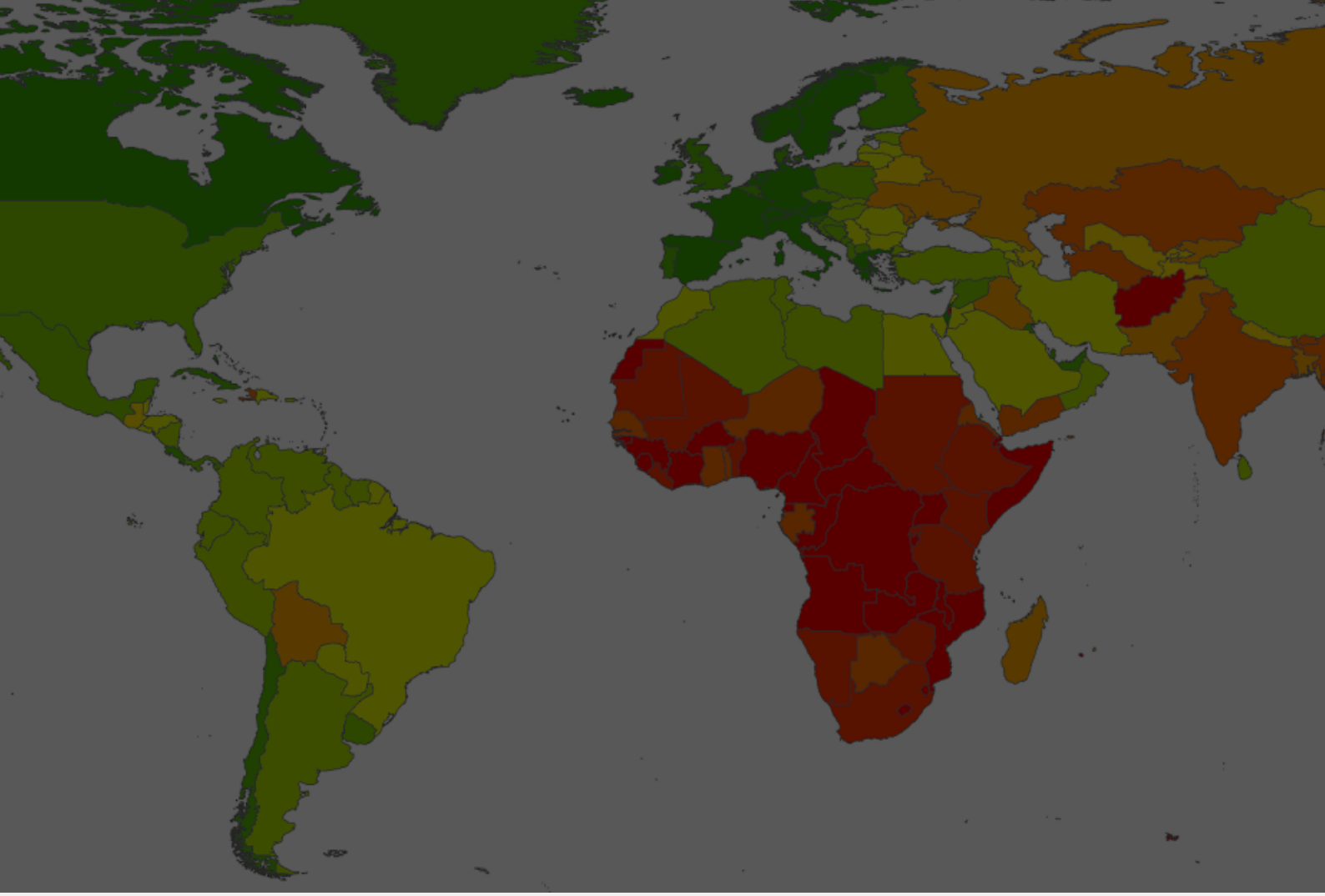

DESIGNING A PREDICTIVE MODEL FOR LIFE EXPECTANCY IN 2020

Written by: 2200634, 2200909

December 14th, 2022

SUMMARY

This project seeks to develop the best linear model to predict mean life expectancy for countries in 2020. It begins with a quantitative and qualitative analysis of the data provided, followed by a descriptive analysis of the methods used when dealing with missing data and collinearity. It then finishes with the presentation of the best predictive model followed by the design of an experiment to test whether there is a difference in mean life expectancy across a categorical predictor (continent).

Contents

Word count (excluding appendix): 2155 words.

1 Introduction

Humans have always been intrigued by the process of life and death. The only definitive aspect of oneself, since the moment one is born, is that they will die; it is here where the fundamental question emerges. This is the question so many people are concerned about, when? If people knew exactly when this was going to happen, they could plan how they would like to spend their time on earth. Obviously, knowing one's moment is an impossible task; however, by looking closely at the data collected in the last few years, it may be possible to predict the average life expectancy of a person given the country they live in. Apart from being an intriguing fact, this information can be beneficial for institutions, such as governments, to make more suitable and informed decisions. This project aims to determine which variables might help accomplish this goal by first describing the data, treating the problem of missing data, and investigating potential issues such as collinearity. It then suggests the best linear model that predicts life expectancy in 2020. Finally, it demonstrates an appropriate experimental design to study differences in average life expectancies across the continents.

1.1 Descriptive statistics

The analysis for this project was conducted on a dataset containing information on 27 features for 217 unique countries. Included among the 27 features were variables covering a wide range economic, demographic, educational, and health-related areas. Table 1 displays means and standard deviations for the *complete cases* of a subset of key variables covering the previously mentioned areas. It is important to note that some variables were missing at a high rate, which will be described and addressed later. This table also shows the distribution of countries among continents.

The variable of interest in this analysis is life expectancy measured in years. Among the 217 countries in the dataset, 19 (8.8%) were missing the life expectancy variable (and thus not included in table 1). Following with the analysis, figure 1 shows the distribution of life expectancy for the complete cases. The histogram A in figure 1 shows that the life expectancy data are left skewed. The maximum life expectancy is 85.08 years (Hong Kong) and the minimum is 53.28 years (Central African Republic). Figure 1B shows that on average, countries that spend a higher percentage of their GDPs (gross domestic products) on healthcare tend to have higher life expectancies. Figure 1C suggests that average life expectancy also increases with GDP per capita, while there is no obvious relationship between life expectancy and population size.

Continent	Number of Countries	Life Expectancy at Birth (years)	Primary School Completion Rate (total of % by age)	Health Expenditure (current international $)	Population Density (people per sq. km)	GDP per Capita (current international $)
Africa	54	64.1 (5.9)	79.8 (15.4)	128.8 (173.4)	104.8 (133.2)	2630.1 (3096.2)
Asia	50	74.6 (5.1)	97.7 (9.8)	721.8 (991.3)	949.5 (3069.1)	15116.8 (19922.9)
Australia/Oceania	19	73.5 (6.0)	94.1 (15.4)	1164.2 (1679.5)	145.2 (152.8)	14783.1 (16508.8)
Europe	48	79.3 (3.6)	98.4 (5.5)	2929.8 (2382.9)	650.7 (2819.2)	40646.5 (40399.5)
North America	34	76.2 (3.8)	92.3 (10.5)	1293.0 (2334.5)	281.6 (295.5)	24082.3 (27212.6)
South America	12	75.1 (3.2)	97.3 (6.8)	675.5 (450.6)	24.7 (18.4)	8495.9 (4220.4)
Total	217	72.9 (7.5)	93.0 (12.9)	1143.7 (1838.1)	446.0 (1996.6)	18605.5 (27774.1)

Table 1. Mean (SD) for subset of variables in life expectancy dataset. Value reported from complete cases only. Missingness present for each variable at the following rates: Life Expectancy 8.8%; Primary Education Completion 41.0%; Health Expenditure 14.3%; Population Density 0.5%; GDP 5.5%.

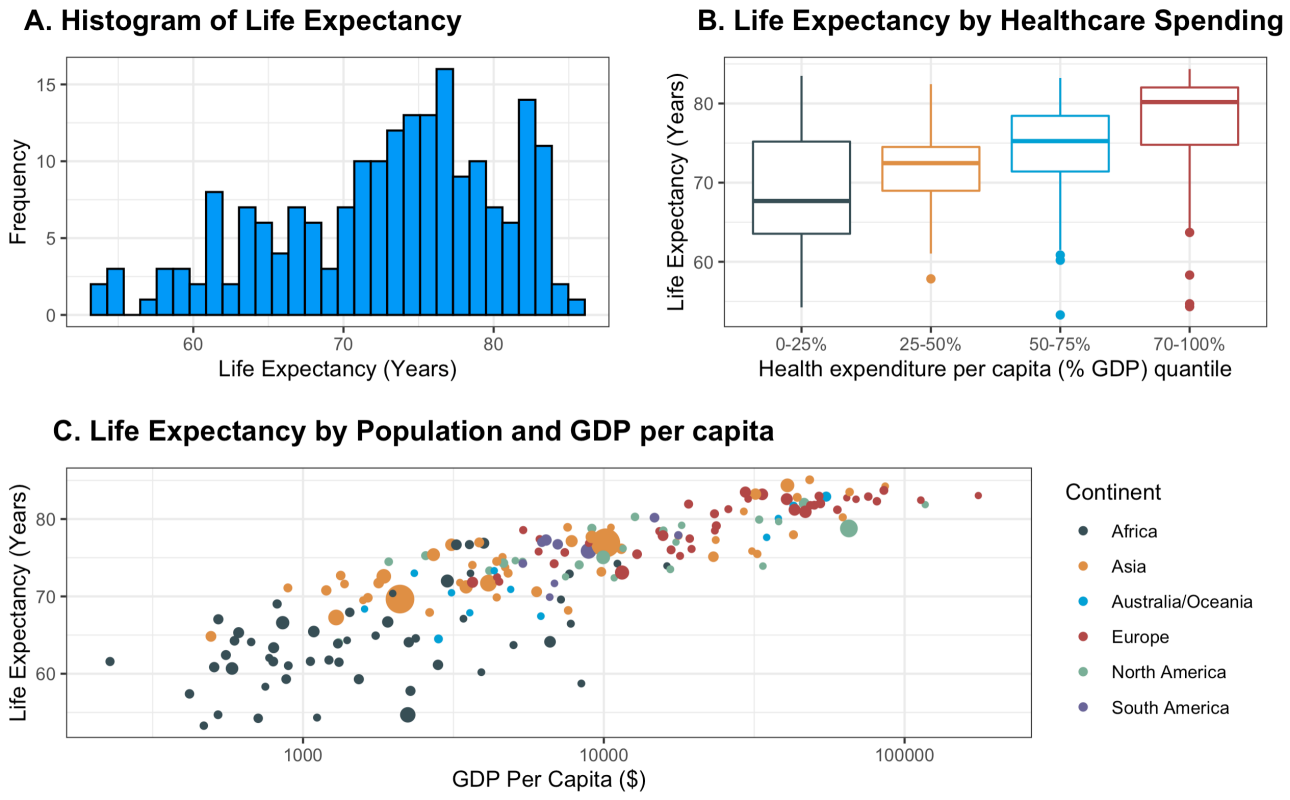

Figure 1: Graphs describing life expectancy. Point size in (C) is proportional to total population.

1.2 Variable transformations

An important detail revealed in Figure 1C is that the relationship between GDP per capita and life expectancy is log-linear. Figure S1 shows that life expectancy is also linearly related the to the log of health expenditure per capita Thus, these two variables were log-transformed for the rest of the analysis.

1.3 Handling missing values

A number of variables contained missing data at high rates. In fact, six had missing entries for over 50% of countries (Table 2). These six features were removed before imputation and ensuing analysis because their extreme high rates of missingness would add only a small amount of information at the expense of increased uncertainty, complexity, and potential bias in the imputation model. Further, these variables closely resembled one or more variables with lower rates of missingness, providing additional justification for their removal. Although some other features were missing at rates from 30%-40%, these features were kept for the multiple imputation model and ensuing analysis based on recommendations in the literature stating that multiple imputation is generally better than ignoring data altogether[1]. One key step taken in this analysis during imputation was to adjust the predictor matrix to remove life expectancy as a predictor for any of the other features. This step prevents baking a direct predictor-response relationship into the data and also keeps the imputation applicable for future predictions where the response variable, life expectancy, would clearly not be available a priori.

Variables	Missing
Children newly infected with HIV	58.5%
Educational attainment, primary	83.4%
Educational attainment, bachelor's	82.5%
Literacy rate	88.5%
Poverty ratio	89.9%
Renewable energy consumption	100%

Table 2. Percent missing for five variables with greatest missingness

After the six variables in Table 2 were removed from the dataset, the rest of the missing data was addressed using multiple imputation. Specifically, multiple imputation by chained equations was employed using the "Mice()" package in R [2]. Following guidance from package documentation, this method was implemented with 10 imputations using predictive mean

matching, a robust technique that utilizes an implicit model to impute values based on the actual distribution of available data points [3]. Importantly, the 19 countries with missing life expectancy were kept in the imputation model solely for leveraging their feature data for feature imputation. These countries were omitted during model development. Figure 2 provides a visual representation of the first five imputations on a subset of features.

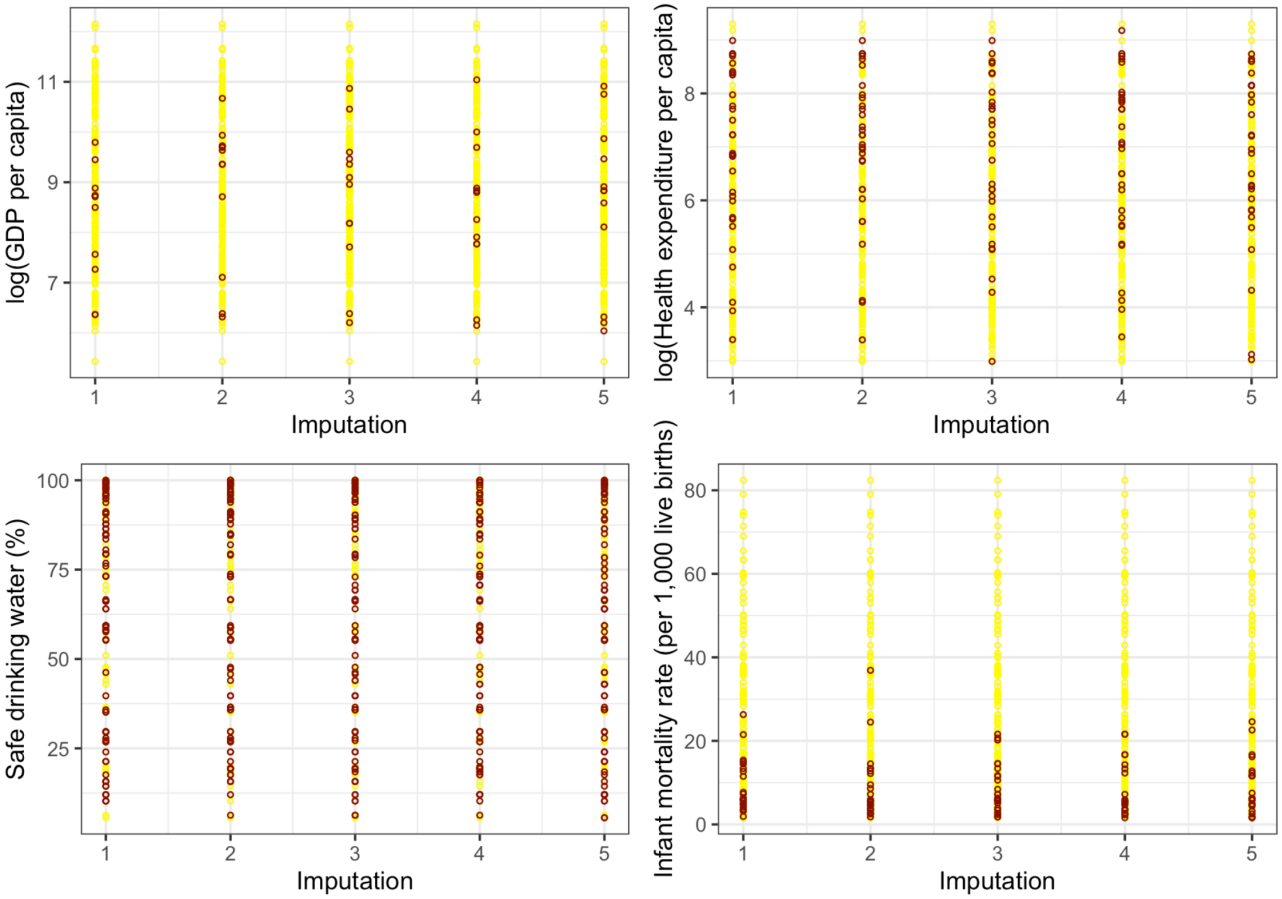

Figure 2: Visual representation of imputations for a subset of features in the first five imputations. Red circles represent imputed values, while yellow circles represent values provided in dataset.

It is important to remember that this method of multiple imputation relies heavily on the assumption that the data are missing at random (MAR). In other words, it hinges on the assumption that the probability of a value being missing only depends on variables which are observed in the data in order for the imputation and ensuing models to be correct. Based on the breadth of variables in the dataset, this assumption appears reasonable.

2 Investigating collinearity

One downside of the large number of features in the dataset is the potential for issues relating to collinearity. Collinearity occurs when multiple features are highly correlated and, if left unaddressed, it can lead to models with inflated variances and hinder statistical inference. To address this, the pairwise correlations between features (Figure 3) were analyzed in addition to the variance inflation factors (VIFs; Appendix Table S1). These results are presented in Figure 3.

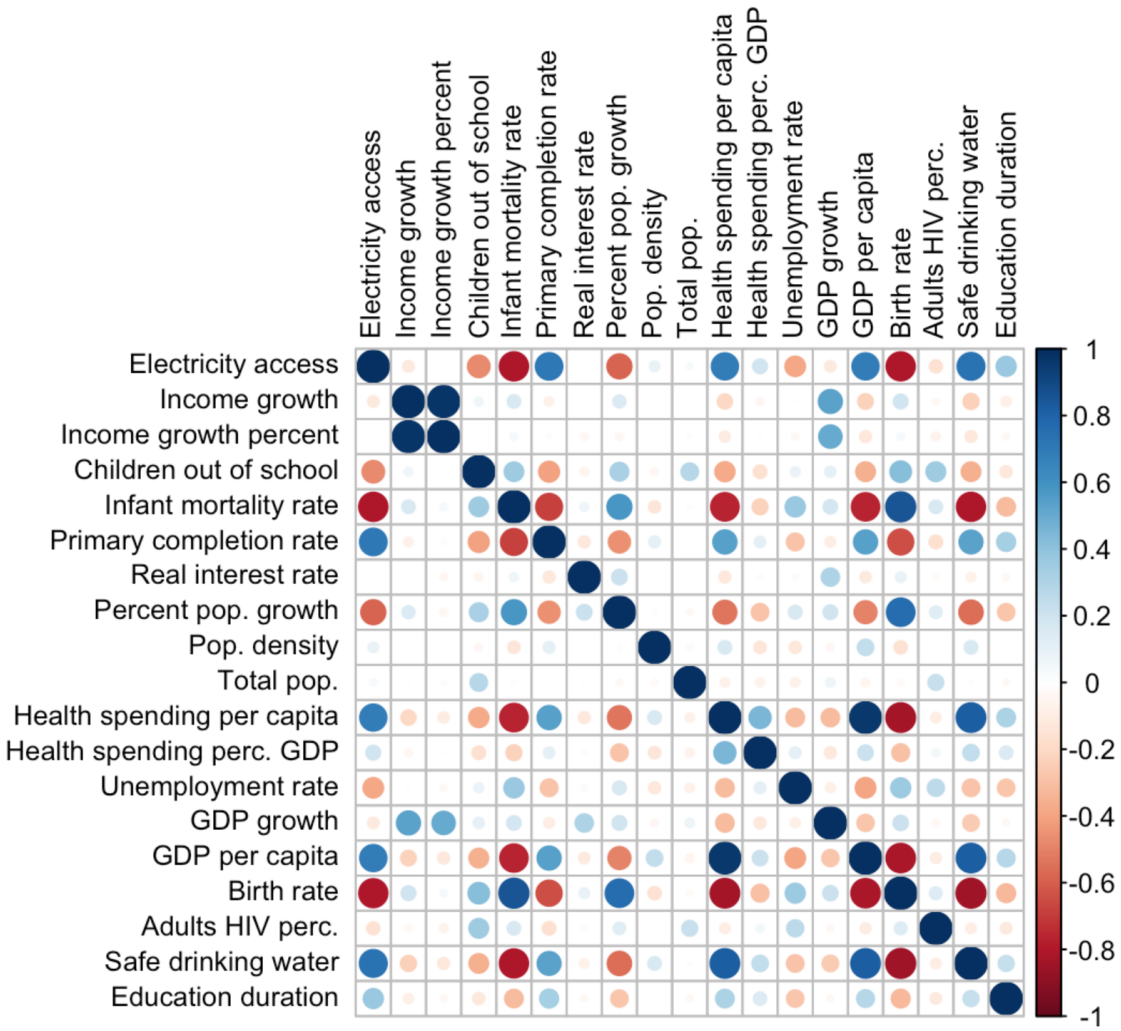

Figure 3: Correlation plot.

This analysis revealed a number of variables that have high VIFs and correlations with absolute values > 0.8. Based on these observations, national net income growth because of its high correlation with national net income growth per capita ($r = 0.979$), birth rate because of its high positive correlation with population growth rate ($r = 0.76$), infant mortality rate ($r =$

0.86) and large negative correlation with safe drinking water rate (r = - 0.85) were removed. Removing these helped to address the issue of multicollinearity which makes the coefficients and their standard errors more appropriate. Electricity access was also removed because of its high VIF and large correlations with infant mortality rate (r = 0.81), primary completion rate (r = 0.71), and safe drinking water rate (r = 0.72). After removing these four variables, all VIFs fall below 5 (Table S1). The next step was then to determine the best linear model.

3 A linear model to predict life expectancy

3.1 Building the model

To find the best model to predict life expectancy in 2020, a forward stepwise regression technique was implemented on the 10 imputed datasets using the "stats()" R package. Importantly, the model was only run over countries that were not initially missing the life expectancy variable, leaving the 19 that were missing it to be predicted later. This procedure helped to determine a best model on each dataset. The proportion of times each feature appeared best subset of features during this process was then analyzed. Five features (Table 3) appeared in 100% of the subsets. A model with these five features as predictors for life expectancy was then fit on each imputed dataset and results were pooled (Table 3). The pooled R-squared value of this model was 89%, suggesting that a large portion of the variance in life expectancy is explained by these predictors. After observing that adding other features to the model (those that appeared in less than 90% of subsets from stepwise regression) did not noticeably improve the adjusted R^2, this prediction model was determined to be the best for life expectancy.

Feature	Estimate	Standard Error	df	P-value	Percent of Best Subsets Appeared in	VIF
Intercept	64.3236	2.05888	105.07	< 0.0001		
Health expenditure (% GDP)	0.1720	0.07885	114.12	0.0292	100%	1.11
Infant mortality rate	-0.2481	0.01687	126.45	< 0.0001	100%	3.09
% Drinking safe water	0.0207	0.01377	51.79	0.1395	100%	4.18
Population density (per km^2)	0.0003	0.00012	99.21	0.0069	100%	1.11
log(GDP per capita)	1.2622	0.25856	75.83	< 0.0001	100%	3.68

Table 3. Final model summary. Results are pooled across m = 10 imputations.

In this final model, four of the five features are statistically significantly associated with life

expectancy at the 5% level. One feature, % of population drinking safe water, is not significant at the 5% level, but it has a p-value of 0.14, appeared in 100% of the best subsets from the stepwise regression, and removing it from the model decreases the adjusted R^2. Since the aim of this model is prediction, this term was kept in the final best model despite its p-value.

The coefficients, errors, and p-values are displayed in the Table 3. After controlling for the other features, the a one unit increase in infant mortality rate is associated with a -0.248 year increase (0.248 year decrease) in life expectancy on average. The other coefficient estimates can be interpreted in a similar manner. Altogether, the model provides evidence that the number of adults newly infected with HIV and the infant mortality rate (possible proxy variables for weaker health system) are associated with lower life expectancies on average, while increased percents of populations using safe drinking water, and increased GDPs per capita are associated with higher life expectancies on average.

To ensure that this is an appropriate final model, plots of the residuals were created (Figure 4). The residuals and fitted values used in this figure are the respective means across the imputations. Figure 4 illustrates that the residuals are approximately normally distributed and shows that there is no clear pattern. Together, these results reassure that the linearity, homoscedasticity, and multivariate normality assumptions for linear regression are met. When combined with the low variance inflation factors, these findings suggest that the proposed model is in fact a good model for life expectancy.

3.2 Predicting life expectancy in other countries

To demonstrate how this linear prediction model can be implemented, predictions for life expectancy were generated for the 19 countries that were missing the variable. These countries were excluded from the model building steps, but they were included in the imputation (of features) steps. Thus, predictions were made using the best model detailed above on the imputed datasets because some of the countries were initially missing predictor values. Table S2 in the appendix shows the mean predicted life expectancy across the 10 imputations for these 19 countries.

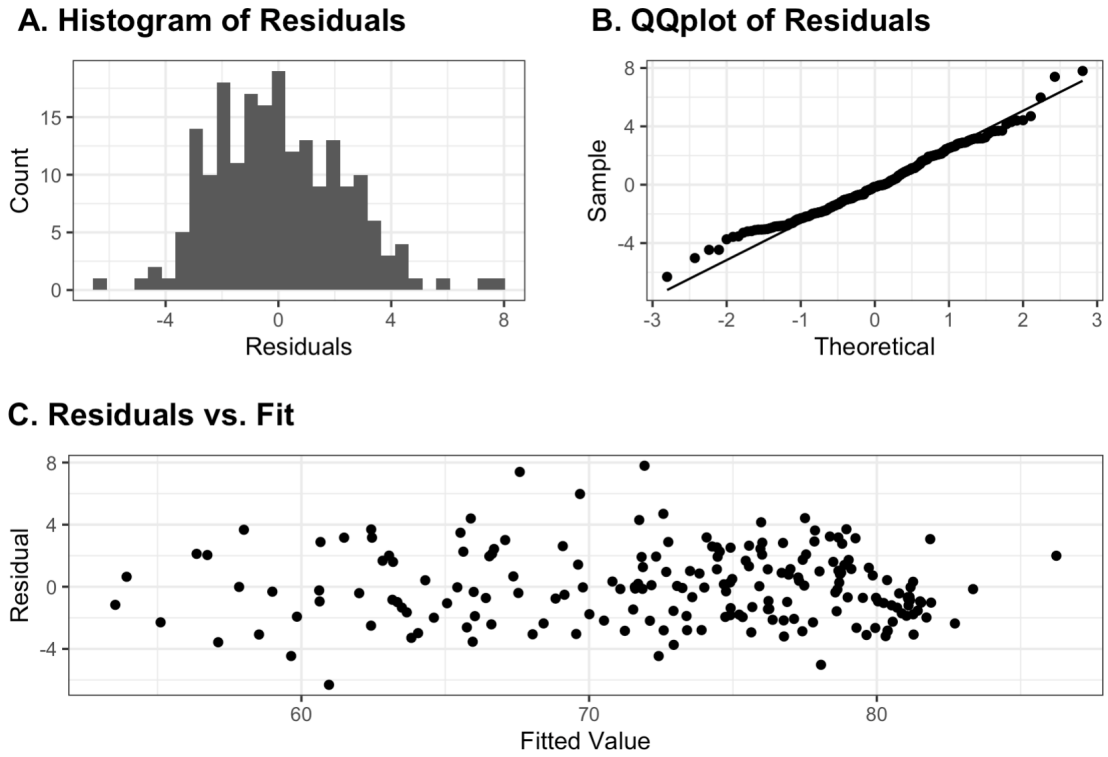

Figure 4: Model assumption check for chosen model

4 Experimental design to study life expectancy across continents

4.1 Proposed Experiment: ANCOVA

To study the differences in life expectancy across continents (a factor variable), an ANCOVA experimental design was employed. ANCOVA is designed specifically for testing whether there is a difference in means for a continuous response variable (in this case, life expectancy) across a categorical predictoor (continent), and it has the additional benefit of controlling for covariates that are independent of the predictor. In the experimental design, an ANCOVA was performed with GDP per capita as a covariate [4]. In both the plots of the data (Figure 1, Figure 5) and in the best model (Table 3), there was a clear association between GDP per capita and life expectancy. Thus, they were used as covariates in the experimental design to ask the question whether there is a difference in mean life expectancy across continent even after adjusting for GDP.

Due to the fact that the references and software for performing ANCOVA and related post-hoc analyses on multiply imputed data are limited, single imputation by chained equations

(multiple imputation with m = 1) was used for this experiment. To combat the limitations of only using one imputation, sensitivity analyses with different random seeds given to the imputation algorithm were also performed to ensure that was not do to the randomness of the single imputation.

4.2 Results and Assumption Verification

The results of the ANCOVA are presented in Table 4

Source of Variation	Sum of Squares	df	Mean Squares	F	p-value
Log(GDP per capita)	7682	1	7682	581.82	< 0.0001
Continent	1190	5	238	18.03	< 0.0001
Residuals	2273	2273	13		
Total	11145	2279			

Table 4. ANCOVA Results

Table 4 shows that the p-value for the F-statistic for Continent is < 0.0001, suggesting that there is a statistically significant difference in mean life expectancy across continents, even after adjusting for log(GDP per capita).

To justify the choice of ANCOVA, the following assumptions for ANCOVA were assessed:

- Normality of residuals
- Equal variances between the different continents
- Homogeneity of regression slopes.
- The relationship between the response and the covariate is linear.

Figure 5A shows that that after log-transforming GDP per capita, it appears to have a linear relationship with life expectancy that is homogenous across continents. Further, the variances look similar across continents. Figure 5B shows that there appear to be some differences in life expectancy by continent, and the variance looks fairly similar across continents.

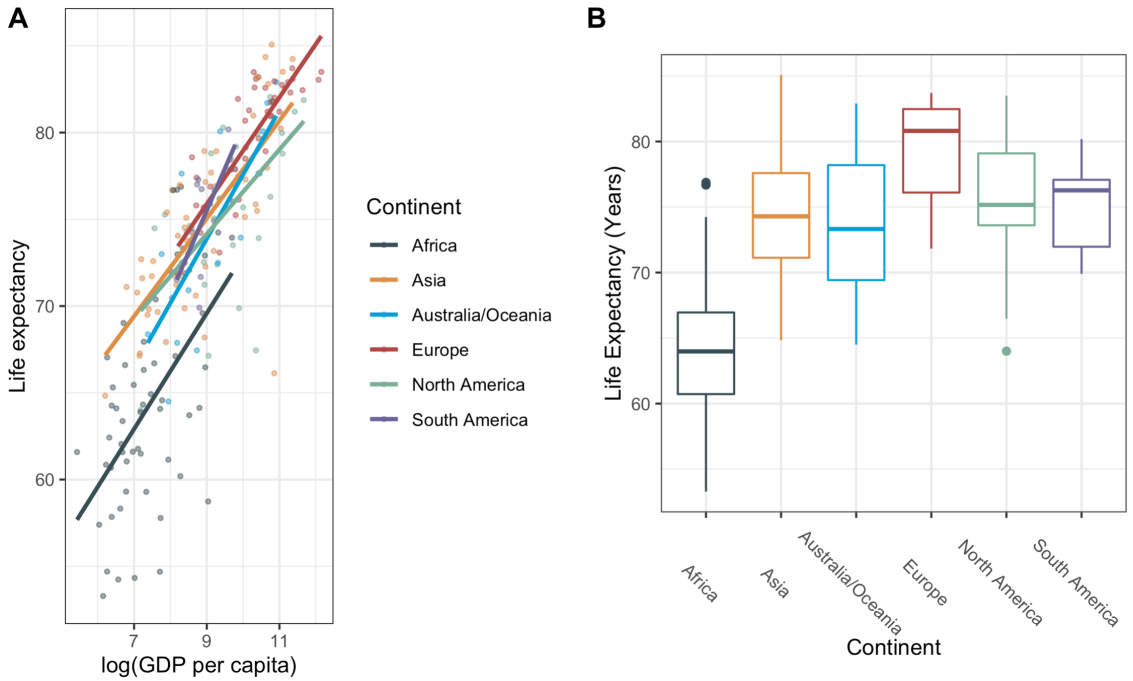

Figure 5: Life expectancy across continents

Figure S1 shows that the residuals appear to be normal. This observation was verified by a Shapiro-Wilk normality test ($p = 0.49$), which did not provide enough evidence to say that the the residuals were not normally distributed. Levine's test for homogeneity of variances was also conducted ($p = 0.046$). This result was marginally significant at the 0.05 level, so the results should be interpreted with some caution.

5 Conclusion

This report has described the methods and results used to create a linear model to predict life expectancy among countries and compare life expectancy across countries in different continents. A linear model using health expenditure (as % of GDP), infant mortality rate, safe drinking water %, population density, and log(GDP per capita) as predictors was found to explain 89.0% of the variance in life expectancy in the dataset. It was then shown that even after controlling for log(GDP per capita), there is evidence that mean life expectancy differs across continents (below 5% significance level). It is important to note that although the models described in this report were shown to be useful for *prediction* the observational nature of data collection means that the findings *cannot be interpreted as causal*.

6 References

[1] Janssen, K. J., Donders, A. R. T., Harrell Jr, F. E., Vergouwe, Y., Chen, Q., Grobbee, D. E., & Moons, K. G. (2010). *Missing covariate data in medical research: to impute is better than to ignore.* Journal of clinical epidemiology, 63(7), 721-727.

[2] van Buuren, S., Groothuis-Oudshoorn, K., Robitzsch, A., Vink, G., Doove, L., & Jolani, S. (2015). Package 'mice'. Computer software.

[3] Van Buuren, S. (2018). *Flexible imputation of missing data.* CRC press.

[4] *R-bloggers (2021). How to perform ANCOVA in R.* Available at: emphhttps://www.r-bloggers.com/2021/07/how-to-perform-ancova-in-r/

7 Appendix

7.1 Supplemental Tables

Feature	VIF Before Dropping Vars	VIF After Dropping Vars
Electricity access	4.63	Dropped
Income growth	1812.86	Dropped
Income growth percent	1788.74	1.61
Children out of school	2.19	1.89
Infant mortality rate	6.25	4.75
Primary completion rate	2.97	2.49
Real interest rate	1.40	1.37
Percent pop. growth	74.53	1.88
Pop. density	3.34	1.38
Total pop.	1.69	1.58
Health spending per capita	10.57	Dropped
Health spending perc. GDP	3.00	1.56
Unemployment rate	2.21	1.78
GDP growth	2.01	1.86
GDP per capita	11.25	1.90
Birth rate	11.26	Dropped
Adults HIV perc.	1.73	1.65
Safe drinking water	5.06	3.94
Education duration	1.49	1.41

Table S1. Variance Inflation Factors (VIF) before and after Dropping variables

Country	Predicted life expectancy (years)
American Samoa	77.11
Andorra	80.19
British Virgin Islands	74.92
Cayman Islands	81.04
Curacao	77.11
Dominica	70.31
Gibraltar	78.36
Greenland	79.11
Isle of Man	80.55
Marshall Islands	72.36
Monaco	87.94
Nauru	73.08
Northern Mariana Islands	77.53
Palau	76.96
San Marino	80.88
Sint Maarten (Dutch part)	75.77
St. Kitts and Nevis	76.55
Turks and Caicos Islands	78.35
Tuvalu	75.57

Table S2. Predictions for countries missing life expectancy variable based on pooled coefficients from best linear model

7.2 Supplemental Figures

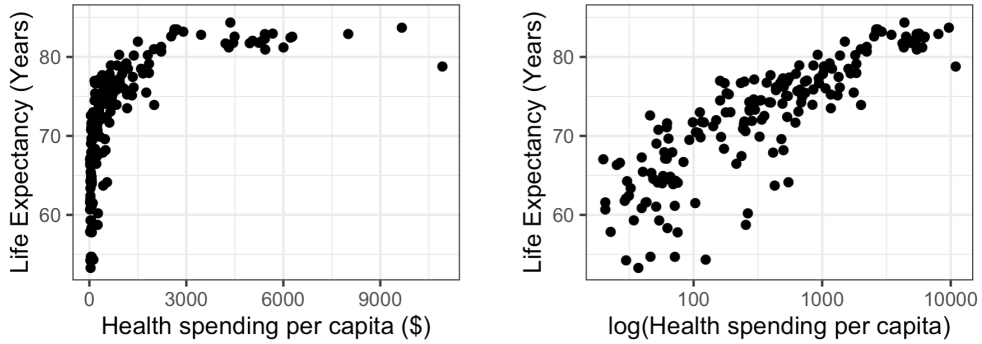

Figure S1: Plots for log transformation of health expenditure per capita variable

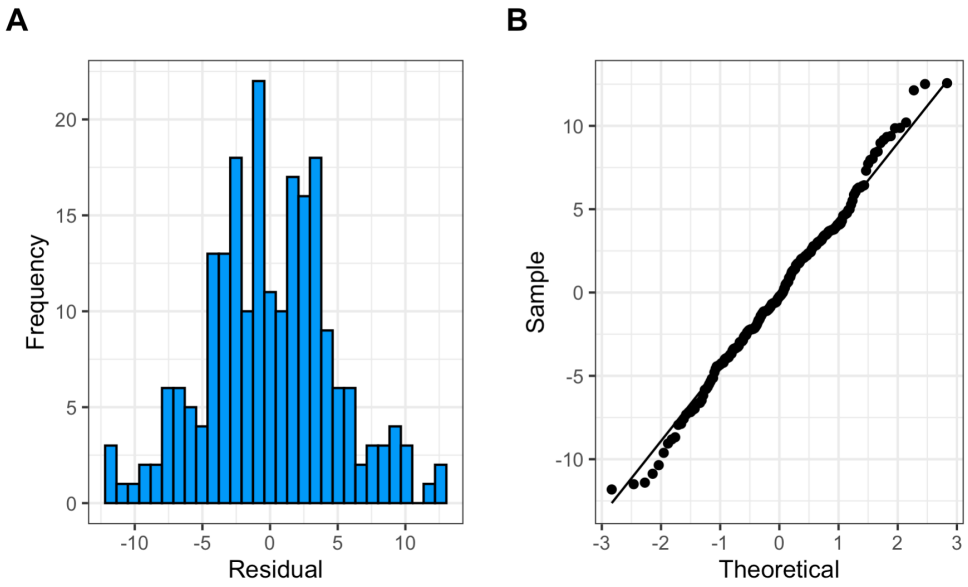

Figure S2: Plots to verify normality of residuals for ANCOVA in Section 4

7.3 R Code to generate results and figures

This section contains the code to replicate the results and figures/tables presented in the report. It is very long, but it does not contain any excess code not used in the report. This code and raw files of the figures are also available at https://github.com/yangjasp/MA317Assessment/blob/main/FinalCode.R.

```
##########
##### Setup ----------------------------
##########

### load packages
library(ISLR)
library(tidyr)
library(dplyr)
library(ggplot2)
library(corrplot)
library(flextable)
library(mice)
library(faraway)
library(cowplot)
library(ggthemes)
library(ggsci)
library(miceadds)
library(officer)
library(multcomp)
```

```r
library(car)

# Read in the data, make initial changes
LE_data_raw <- read.csv("Life_Expectancy_Data1.csv")
LE_data <- subset(LE_data_raw, select = -c(Country.Code,EG.FEC.RNEW.ZS)) #we deleted ..
# Renewable energy consumption because it had 100% missingness.
# And don't need Country Code as it is redundant for Country Name
LE_data$SP.POP.TOTL <- LE_data$SP.POP.TOTL/100000 # Change units of population to be more interpretable
LE_data$SH.HIV.INCD <- LE_data$SH.HIV.INCD/1000 # Change units of HIV adults to 1000s

# Set number of imputations
m <- 10

#########
### Question 1 - Descriptive Statistics and Imputation ----------------
##########

# First, we are going to analyze in broad strokes, the main characteristics of our data.

# N obs
dim(LE_data) # 217 observations, 29 columns

# N missing for LE
table(is.na(LE_data$SP.DYN.LE00.IN))
table(is.na(LE_data$SP.DYN.LE00.IN))/nrow(LE_data) # percent missing

# Max and min values for life expectancy
# Max
max(LE_data$SP.DYN.LE00.IN, na.rm = TRUE)
dplyr::arrange(LE_data, desc(SP.DYN.LE00.IN))$Country.Name[1] # country
# min
min(LE_data$SP.DYN.LE00.IN, na.rm = TRUE)
dplyr::arrange(LE_data, SP.DYN.LE00.IN)$Country.Name[1] # country

# Renaming our variables so it's easier to handle the data
# Names key
new_names <- c("Country.Name", "Continent", "life_expectancy",
               "electricity_access", "income_growth", "income_growth_pc",
               "children_HIV", "children_out_of_school",
               "ed_attain_primary_rate", "ed_attain_bach_rate",
```

```
                "infant_mortality_rate", "primary_completion_rate",
                "literacy_rate", "real_interest_rate", "pop_growth_perc_
    annual",
                "pop_dens", "pop_total", "health_expend_pc",
                "health_expend_perc_gdp", "unemployment_rate", "gdp_growth",
                "gdp_pc", "birth_rate", "adults_HIV", "safe_drinking_water",
                "poverty_ratio", "ed_duration")

names(LE_data) <- new_names

## Table 1: Table by Continents

# Following the small exploration of data we're doing, we chose 4 areas (
    education,
# population density, health and wealth), and of course, life expectancy
    itself,
# on which we'll see how the different continents behave

## Split data by continent for table
LE_Africa <- dplyr::filter(LE_data, Continent == 'Africa')
LE_Asia <- dplyr::filter(LE_data, Continent == 'Asia')
LE_Oceania <- dplyr::filter(LE_data, Continent == 'Australia/Oceania')
LE_Europe <- dplyr::filter(LE_data, Continent == 'Europe')
LE_NA <- dplyr::filter(LE_data, Continent == 'North America')
LE_SA <- dplyr::filter(LE_data, Continent == 'South America')


# Mean and standard deviation of the 5 variables we're studying
# In order for us to visualize the data in a "easier" way, we're going
    summarize it in a table

## Life Expectancy
life_expectancy_Africa <- paste0(
  sprintf("%.1f",mean(LE_Africa$life_expectancy, na.rm = TRUE)), ' (',
    sprintf("%.1f", sd(LE_Africa$life_expectancy, na.rm = TRUE)), ")")
life_expectancy_Asia <- paste0(
  sprintf("%.1f",mean(LE_Asia$life_expectancy, na.rm = TRUE)), ' (', sprintf
    ("%.1f",sd(LE_Asia$life_expectancy, na.rm = TRUE)), ")")
life_expectancy_Oceania <- paste0(
  sprintf("%.1f",mean(LE_Oceania$life_expectancy, na.rm = TRUE)), ' (',
    sprintf("%.1f",sd(LE_Oceania$life_expectancy, na.rm = TRUE)), ")")
life_expectancy_Europe <- paste0(
  sprintf("%.1f",mean(LE_Europe$life_expectancy, na.rm = TRUE)), ' (',
    sprintf("%.1f",sd(LE_Europe$life_expectancy, na.rm = TRUE)), ")")
```

```r
life_expectancy_NA <- paste0(
  sprintf("%.1f",mean(LE_NA$life_expectancy, na.rm = TRUE)), ' (', sprintf("%.1f",sd(LE_NA$life_expectancy, na.rm = TRUE)), ")")
life_expectancy_SA <- paste0(
  sprintf("%.1f",mean(LE_SA$life_expectancy, na.rm = TRUE)), ' (', sprintf("%.1f",sd(LE_SA$life_expectancy, na.rm = TRUE)), ")")
life_expectancy_All <- paste0(
  sprintf("%.1f",mean(LE_data$life_expectancy, na.rm = TRUE)), ' (', sprintf("%.1f",sd(LE_data$life_expectancy, na.rm = TRUE)), ")")
# Combine it to a vector
table1_life_expectancy <- c(life_expectancy_Africa,
                            life_expectancy_Asia,
                            life_expectancy_Oceania,
                            life_expectancy_Europe,
                            life_expectancy_NA,
                            life_expectancy_SA,
                            life_expectancy_All)

## Generate table values, first Primary Educational Attainment population aged 25+ (units by %)
primary_completion_Africa <- paste0(
  sprintf("%.1f",mean(LE_Africa$primary_completion_rate, na.rm = TRUE)), ' (',
  sprintf("%.1f", sd(LE_Africa$primary_completion_rate, na.rm = TRUE)), ")")
primary_completion_Asia <- paste0(
  sprintf("%.1f",mean(LE_Asia$primary_completion_rate, na.rm = TRUE)), ' (',
  sprintf("%.1f",sd(LE_Asia$primary_completion_rate, na.rm = TRUE)), ")")
primary_completion_Oceania <- paste0(
  sprintf("%.1f",mean(LE_Oceania$primary_completion_rate, na.rm = TRUE)), ' (',
  sprintf("%.1f",sd(LE_Oceania$primary_completion_rate, na.rm = TRUE)), ")")
primary_completion_Europe <- paste0(
  sprintf("%.1f",mean(LE_Europe$primary_completion_rate, na.rm = TRUE)), ' (',
  sprintf("%.1f",sd(LE_Europe$primary_completion_rate, na.rm = TRUE)), ")")
primary_completion_NA <- paste0(
  sprintf("%.1f",mean(LE_NA$primary_completion_rate, na.rm = TRUE)), ' (',
  sprintf("%.1f",sd(LE_NA$primary_completion_rate, na.rm = TRUE)), ")")
primary_completion_SA <- paste0(
  sprintf("%.1f",mean(LE_SA$primary_completion_rate, na.rm = TRUE)), ' (',
  sprintf("%.1f",sd(LE_SA$primary_completion_rate, na.rm = TRUE)), ")")
primary_completion_All <- paste0(
  sprintf("%.1f",mean(LE_data$primary_completion_rate, na.rm = TRUE)), ' (',
  sprintf("%.1f",sd(LE_data$primary_completion_rate, na.rm = TRUE)), ")")
```

```r
# Combine it to a vector
table1_primary_completion <- c(primary_completion_Africa,
                               primary_completion_Asia,
                               primary_completion_Oceania,
                               primary_completion_Europe,
                               primary_completion_NA,
                               primary_completion_SA,
                               primary_completion_All)

## Population Density
pop_density_Africa <- paste0(
  sprintf("%.1f",mean(LE_Africa$pop_dens, na.rm = TRUE)), ' (',
  sprintf("%.1f", sd(LE_Africa$pop_dens, na.rm = TRUE)), ")")
pop_density_Asia <- paste0(
  sprintf("%.1f",mean(LE_Asia$pop_dens, na.rm = TRUE)), ' (',
  sprintf("%.1f",sd(LE_Asia$pop_dens, na.rm = TRUE)), ")")
pop_density_Oceania <- paste0(
  sprintf("%.1f",mean(LE_Oceania$pop_dens, na.rm = TRUE)), ' (',
  sprintf("%.1f",sd(LE_Oceania$pop_dens, na.rm = TRUE)), ")")
pop_density_Europe <- paste0(
  sprintf("%.1f",mean(LE_Europe$pop_dens, na.rm = TRUE)), ' (',
  sprintf("%.1f",sd(LE_Europe$pop_dens, na.rm = TRUE)), ")")
pop_density_NA <- paste0(
  sprintf("%.1f",mean(LE_NA$pop_dens, na.rm = TRUE)), ' (',
  sprintf("%.1f",sd(LE_NA$pop_dens, na.rm = TRUE)), ")")
pop_density_SA <- paste0(
  sprintf("%.1f",mean(LE_SA$pop_dens, na.rm = TRUE)), ' (',
  sprintf("%.1f",sd(LE_SA$pop_dens, na.rm = TRUE)), ")")
pop_density_All <- paste0(
  sprintf("%.1f",mean(LE_data$pop_dens, na.rm = TRUE)), ' (',
  sprintf("%.1f",sd(LE_data$pop_dens, na.rm = TRUE)), ")")
# Combine it to a vector
table1_pop_density <- c(pop_density_Africa,
                        pop_density_Asia,
                        pop_density_Oceania,
                        pop_density_Europe,
                        pop_density_NA,
                        pop_density_SA,
                        pop_density_All)

## Health Expenditure per Capita
health_expenditure_Africa <- paste0(
  sprintf("%.1f",mean(LE_Africa$health_expend_pc, na.rm = TRUE)), ' (',
  sprintf("%.1f", sd(LE_Africa$health_expend_pc, na.rm = TRUE)), ")")
```

```
health_expenditure_Asia <- paste0(
  sprintf("%.1f",mean(LE_Asia$health_expend_pc, na.rm = TRUE)), ' (',
  sprintf("%.1f",sd(LE_Asia$health_expend_pc, na.rm = TRUE)), ")")
health_expenditure_Oceania <- paste0(
  sprintf("%.1f",mean(LE_Oceania$health_expend_pc, na.rm = TRUE)), ' (',
  sprintf("%.1f",sd(LE_Oceania$health_expend_pc, na.rm = TRUE)), ")")
health_expenditure_Europe <- paste0(
  sprintf("%.1f",mean(LE_Europe$health_expend_pc, na.rm = TRUE)), ' (',
  sprintf("%.1f",sd(LE_Europe$health_expend_pc, na.rm = TRUE)), ")")
health_expenditure_NA <- paste0(
  sprintf("%.1f",mean(LE_NA$health_expend_pc, na.rm = TRUE)), ' (',
  sprintf("%.1f",sd(LE_NA$health_expend_pc, na.rm = TRUE)), ")")
health_expenditure_SA <- paste0(
  sprintf("%.1f",mean(LE_SA$health_expend_pc, na.rm = TRUE)), ' (',
  sprintf("%.1f",sd(LE_SA$health_expend_pc, na.rm = TRUE)), ")")
health_expenditure_All <- paste0(
  sprintf("%.1f",mean(LE_data$health_expend_pc, na.rm = TRUE)), ' (',
  sprintf("%.1f",sd(LE_data$health_expend_pc, na.rm = TRUE)), ")")
# Combine it to a vector
table1_health_expenditure <- c(health_expenditure_Africa,
                               health_expenditure_Asia,
                               health_expenditure_Oceania,
                               health_expenditure_Europe,
                               health_expenditure_NA,
                               health_expenditure_SA,
                               health_expenditure_All)

## GDP per Capita
GDP_Africa <- paste0(
  sprintf("%.1f",mean(LE_Africa$gdp_pc, na.rm = TRUE)), ' (',
  sprintf("%.1f", sd(LE_Africa$gdp_pc, na.rm = TRUE)), ")")
GDP_Asia <- paste0(
  sprintf("%.1f",mean(LE_Asia$gdp_pc, na.rm = TRUE)), ' (',
  sprintf("%.1f",sd(LE_Asia$gdp_pc, na.rm = TRUE)), ")")
GDP_Oceania <- paste0(
  sprintf("%.1f",mean(LE_Oceania$gdp_pc, na.rm = TRUE)), ' (',
  sprintf("%.1f",sd(LE_Oceania$gdp_pc, na.rm = TRUE)), ")")
GDP_Europe <- paste0(
  sprintf("%.1f",mean(LE_Europe$gdp_pc, na.rm = TRUE)), ' (',
  sprintf("%.1f",sd(LE_Europe$gdp_pc, na.rm = TRUE)), ")")
GDP_NA <- paste0(
  sprintf("%.1f",mean(LE_NA$gdp_pc, na.rm = TRUE)), ' (',
  sprintf("%.1f",sd(LE_NA$gdp_pc, na.rm = TRUE)), ")")
GDP_SA <- paste0(
```

```
  sprintf("%.1f",mean(LE_SA$gdp_pc, na.rm = TRUE)), ' (',
  sprintf("%.1f",sd(LE_SA$gdp_pc, na.rm = TRUE)), ")")
GDP_All <- paste0(
  sprintf("%.1f",mean(LE_data$gdp_pc, na.rm = TRUE)), ' (',
  sprintf("%.1f",sd(LE_data$gdp_pc, na.rm = TRUE)), ")")
# Combine it to a vector
table1_GDP <- c(GDP_Africa, GDP_Asia, GDP_Oceania, GDP_Europe, GDP_NA, GDP_
    SA, GDP_All)

## Add ncountries
ncountries_Africa <- nrow(LE_Africa)
ncountries_Asia <- nrow(LE_Asia)
ncountries_Oceania <- nrow(LE_Oceania)
ncountries_Europe <- nrow(LE_Europe)
ncountries_NA <- nrow(LE_NA)
ncountries_SA <- nrow(LE_SA)
ncountries_All <- nrow(LE_data)
# Combine it to a vector
table1_ncountries <- c(ncountries_Africa, ncountries_Asia, ncountries_
    Oceania, ncountries_Europe, ncountries_NA, ncountries_SA, ncountries_All)

## Gather missingness % for each variable shown in table (for footnote)
life_expectancy_pmissing <- paste0(
  sprintf("%.1f",mean(is.na(LE_data$life_expectancy))*100), '%')

primary_completion_pmissing <- paste0(
  sprintf("%.1f",mean(is.na(LE_data$primary_completion_rate))*100), '%')

pop_density_pmissing <- paste0(
  sprintf("%.1f",mean(is.na(LE_data$pop_dens))*100), '%')

health_expenditure_pmissing <- paste0(
  sprintf("%.1f",mean(is.na(LE_data$health_expend_pc))*100), '%')

GDP_pmissing <- paste0(
  sprintf("%.1f",mean(is.na(LE_data$gdp_pc))*100), '%')


# Showing results

# Combining all the information
table1_df <- c()
table1_df$Continent <- c("Africa", "Asia", "Australia/Oceania", "Europe", "
    North America", "South America", "Total")
```

```r
table1_df$ncountries <- table1_ncountries
table1_df$Life_expectancy <- table1_life_expectancy
table1_df$primary_completion <- table1_primary_completion
table1_df$health_expenditure <- table1_health_expenditure
table1_df$pop_density <- table1_pop_density
table1_df$GDP <- table1_GDP

# Setting the tittles for the table
table1_df <- as.data.frame(table1_df)
names(table1_df) <- c("Continent",
                      "Number of Countries",
                      "Life Expectancy at Birth (years)",
                      "Primary School Completion Rate (total of % by age)",
                      "Health Expenditure (current international $)",
                      "Population Density (people per sq. km)",
                      "GDP per Capita (current international $)")

# Table properties
table1_table <- flextable(table1_df, col_keys = names(table1_df))

table1_table <- align(table1_table,j =1, align = "left", part = "all")
table1_table <- align(table1_table,j = 2:7, align = "right", part = "all")
table1_table <- width(table1_table, width = c(1.4,1.0,1.0,1.0,1.0,1.0, 1.0))
table1_table <- hrule(table1_table, rule = "at least", part = "all")
table1_table <- hline(table1_table, i = 6, j = NULL, part = "body",
                      border = fp_border(color="black"))

table1_table <- footnote(table1_table, j = 1, i = 1, part="body",  value =
   as_paragraph("Mean (SD)" ))
table1_table <- footnote(table1_table, j = 1, i = 2, part="body",  value =
   as_paragraph(paste0(
  "Value reported from complete cases only. Missingness present for each
   variable at the following rates: ",
  "Life Expectancy ", life_expectancy_pmissing, ";",
  "Primary Education Completion ", primary_completion_pmissing, ";",
  "Health Expenditure ", health_expenditure_pmissing, ";",
  "Population Density ", pop_density_pmissing, ";",
  "GDP ", GDP_pmissing, ".")))

# Final table
table1_table

## Table 2: Variables with Missingness over 50%
```

```r
# We're going to analyse the missingness of our data since we noticed a lot
    is missing

# Loop to get the number of missing values per variable
x <- c()
for (i in 1:ncol(LE_data))
{
  x[i] = mean(is.na(LE_data[,i])) # quantity
  names(x)[i] = names(LE_data)[i] # variable
}

# Split the data into low and high missing values
# Low
low_missing <- x[x < 0.5] # using threshold of 50% missingness
low_missing_vars = names(low_missing)
# High
high_missing <- x[x >= 0.5]
high_missing_vars <- names(high_missing)

# Writing the correct title names. Plus, note we also manually added the
# variable with 100% missingness to the table after generation
high_missing_vars_clean <- c("Children newly infected with HIV",
                             "Educational attainment, primary",
                             "Educational attainment, bachelor's",
                             "Literacy rate", "Poverty Ratio")

# Loop to get table values
high_missing_char <- c()
for (i in 1:length(high_missing))
{
  high_missing_char[i] <- paste0(sprintf("%.1f", high_missing[i]*100),"%")
}

# Combining all the information
table2_df <- c()
table2_df$Var <- high_missing_vars_clean
table2_df$nmissing <- high_missing_char

# Setting the tittles for the table
table2_df <- as.data.frame(table2_df)
names(table2_df) <- c("Variables", "Missingness")

# Table properties
table2_table <- flextable(table2_df, col_keys = names(table2_df))
```

```
345
346 table2_table <- align(table2_table,j =1, align = "left", part = "all")
347 table2_table <- align(table2_table,j = 2, align = "right", part = "all")
348 table2_table <- width(table2_table, width = c(1.4,1.4))
349 table2_table <- hrule(table2_table, rule = "at least", part = "all")
350
351 # Final table
352 table2_table
353
354 #####
355 ## Figure 1: graphs for out variable of interest: LE
356 #####
357
358 # Continuing with the exploratory analysis, we are plotting LE in different
    ways
359
360 ## Plot 1: Histogram of life expectancy
361 p1 <- ggplot(LE_data, aes(life_expectancy)) +
362   geom_histogram(color = "#000000", fill = "#0099F8") +
363   theme_bw()+
364   xlab("Life Expectancy (Years)")+
365   ylab("Frequency")+
366   theme(plot.margin = unit(c(1,0.5,0.5,0.5), "cm"))
367
368 ## Plot 2: boxplot by Health Spending quantiles
369 # Setting the quantiles
370 health_expend_quantiles <- quantile(LE_data$health_expend_perc_gdp, c(0.25,
    0.5, 0.75), na.rm = TRUE)
371 temp <- LE_data %>%
372   dplyr::mutate(HE_quantile = case_when(
373     health_expend_perc_gdp < health_expend_quantiles[1] ~ "0-25%",
374     health_expend_perc_gdp < health_expend_quantiles[2] ~ "25-50%",
375     health_expend_perc_gdp < health_expend_quantiles[3] ~ "50-75%",
376     health_expend_perc_gdp >= health_expend_quantiles[3] ~ "70-100%")) %>%
377   dplyr::filter(!(is.na(health_expend_perc_gdp)))
378 # Plotting
379 p2 <- ggplot(temp, aes(x = HE_quantile, y = life_expectancy,
380                        color = HE_quantile)) +
381   geom_boxplot(show.legend = FALSE) +
382   ggsci::scale_color_jama()+
383   theme(axis.text.x = element_text(angle = 45))+
384   theme_bw()+
385   xlab("Health expenditure per capita (% GDP) quantile")+
386   ylab("Life Expectancy (Years)") +
```

```
  theme(plot.margin = unit(c(1,0.5,0.5,0.5), "cm"))

## Plot 3: Healthcare per capita (plus visualization of size per continent)
p3 <- ggplot(data = LE_data, aes(x = gdp_pc,
                                 y = life_expectancy))+
  geom_point(aes(size = pop_total, color = Continent),
             show.legend = c(TRUE))  +
  ggsci::scale_color_jama()+
  scale_x_log10() +
  theme_bw()+
  ylab("Life Expectancy (Years)") +
  xlab("GDP Per Capita ($)") +
  guides(size = "none") +
  theme(plot.margin = unit(c(1,0.5,0.5,0.5), "cm"))

# Plot in a grid with cowplot::plot_grid() and also give titles
top <- cowplot::plot_grid(p1, p2, labels = c("A. Histogram of Life
    Expectancy", "B. Life Expectancy by Healthcare Spending"),
                          hjust = -0.1)
bottom <- cowplot::plot_grid(p3, labels = c("C. Life Expectancy by
    Population and GDP per capita"),
                             hjust = -0.1)
cowplot::plot_grid(top, bottom, ncol = 1)

#####
### Figure S1: Life expectancy by health expenditure per capita
######
# Also plot life exp, health expenditure per capita for Figure S1 (to show
    we need to log-transform)
p1 <- ggplot(data = LE_data, aes(x = health_expend_pc,
                                 y = life_expectancy))+
  geom_point()  +
  ggsci::scale_color_jama()+
  theme_bw()+
  ylab("Life Expectancy (Years)") +
  xlab("Health spending per capita ($)") +
  guides(size = "none") +
  theme(plot.margin = unit(c(1,0.5,0.5,0.5), "cm"))
p2 <- ggplot(data = LE_data, aes(x = health_expend_pc,
                                 y = life_expectancy))+
  geom_point()  +
  ggsci::scale_color_jama()+
  scale_x_log10() +
  theme_bw()+
```

```
  ylab("Life Expectancy (Years)") +
  xlab("log(Health spending per capita)") +
  guides(size = "none") +
  theme(plot.margin = unit(c(1,0.5,0.5,0.5), "cm"))

cowplot::plot_grid(p1,p2,nrow = 1)

# Log-transform GDP per capita and health_expend_pc in dataset
LE_data$gdp_pc <- log(LE_data$gdp_pc)
LE_data$health_expend_pc <- log(LE_data$health_expend_pc)

####
## Section 1B: Missing Values -------------------
####

# Subset variables with missingness < 50% (we identified these vars earlier)
LE_data1 <- LE_data[ ,names(LE_data) %in% low_missing_vars]
Country.Name <- LE_data$Country.Name

# Remove ID column to get final dataset for imputation
LE_data2 <- subset(LE_data1, select = -c(Country.Name))

# Now impute using using the mice package, which performs multiple
    imputation by chained equations.
#?mice() to access documentation

# First, we create predictor matrix without LE, so LE isn't used to predict
    covariates.
pred <- mice::make.predictorMatrix(LE_data2)
pred[,"life_expectancy"] <- 0 # never use it as a predictor

# Imputation using method predictive mean matching (pmm) and m = 10
    iterations
imputations <- mice(LE_data2, seed = 98, predictorMatrix = pred, method = "
    pmm", m = 10)

#######
#### Figure 2: Plot of Imputations
#######

# stipplot() as we used in lab did not seem to work for some reason.
# So we plot the imputed and observed values for a subset of features
    manually.
```

```
# For this, we just chose a somewhat random subset of features that covered different
# levels of missingness

# Countries missing life expectancy data
countries_missing_gdp_pc <- dplyr::filter(LE_data, is.na(gdp_pc))$Country.Name
countries_missing_health_expend_pc <- dplyr::filter(LE_data, is.na(health_expend_pc))$Country.Name
countries_missing_safe_drinking_water <- dplyr::filter(LE_data,
                                                       is.na(safe_drinking_water))$Country.Name
countries_missing_infant_mortality_rate <- dplyr::filter(LE_data,
                                                         is.na(infant_mortality_rate))$Country.Name

# We will only plot first 5 imputations so it is easy to see. Extract first 5 imputed datasets
vars <- c("life_expectancy", "adults_HIV", "health_expend_pc", "infant_mortality_rate")
imp1 <- cbind(LE_data$Country.Name,"imp" = 1, complete(imputations, 1))
imp2 <- cbind(LE_data$Country.Name,"imp" = 2, complete(imputations, 2))
imp3 <- cbind(LE_data$Country.Name,"imp" = 3, complete(imputations, 3))
imp4 <- cbind(LE_data$Country.Name,"imp" = 4, complete(imputations, 4))
imp5 <- cbind(LE_data$Country.Name,"imp" = 5, complete(imputations, 5))
imps <- rbind(imp1, imp2, imp3, imp4, imp5)

# Re-add country name, which we had removed before imputing.
# Also define new variable specifying which points were imputed
names(imps)[1] <- "Country.Name"
imps <- imps %>%
  dplyr::mutate(Type_gdp = ifelse(Country.Name %in% countries_missing_gdp_pc,
                                  "Imputed", "Not Imputed"),
                Type_safe_drinking = ifelse(Country.Name %in% countries_missing_safe_drinking_water,
                                            "Imputed", "Not Imputed"),
                Type_health = ifelse(Country.Name %in% countries_missing_health_expend_pc,
                                     "Imputed", "Not Imputed"),
                Type_infant = ifelse(Country.Name %in% countries_missing_infant_mortality_rate,
                                     "Imputed", "Not Imputed"))

```

```
p1 <- ggplot(arrange(imps, desc(Type_gdp)), aes(x = imp,
                                                y = gdp_pc,
                                                color = Type_gdp)) +
  geom_point(size = 0.8, shape = 1,
             show.legend = FALSE) +
  theme_bw()+
  scale_color_manual(values = c("#8B0000", "#FFFF00")) +
  xlab("Imputation") +
  ylab("log(GDP per capita)")

p3 <- ggplot(arrange(imps, desc(Type_safe_drinking)), aes(x = imp,
                                                          y = safe_drinking_water,
                                                          color = Type_safe_drinking)) +
  geom_point(shape = 1, size = 0.8, show.legend = FALSE) +
  theme_bw()+
  scale_color_manual(values = c("#8B0000", "#FFFF00")) +
  xlab("Imputation") +
  ylab("Safe drinking water (%)")

p2 <- ggplot(arrange(imps, desc(Type_health)), aes(x = imp,
                                                   y = health_expend_pc,
                                                   color = Type_health)) +
  geom_point(shape = 1, size = 0.8, show.legend = FALSE) +
  theme_bw()+
  scale_color_manual(values = c("#8B0000", "#FFFF00")) +
  xlab("Imputation") +
  ylab("log(Health expenditure per capita)")

p4 <- ggplot(arrange(imps, desc(Type_infant)), aes(x = imp,
                                                   y = infant_mortality_rate,
                                                   color = Type_infant)) +
  geom_point(shape = 1, size = 0.8, show.legend = FALSE) +
  theme(axis.title.y = element_text(size = 9)) +
  theme_bw()+
  scale_color_manual(values = c("#8B0000", "#FFFF00")) +
  xlab("Imputation") +
  ylab("Infant mortality rate (per 1,000 live births)")

cowplot::plot_grid(p1, p2, p3, p4, nrow = 2, hjust = -0.1)

#######
```

```r
#### Question 2: Dealing with Collinearity ---------------------
#######

# Studying collinearity presence, which we can do on multiply imputed data using
# the 'miceadds' package
#?miceadds

# First, we are calculating correlations by removing Continent (factor) and life expectancy (response)
mi_corr <- micombine.cor(imputations, variables = c(3:21))

# This gives pairwise correlations in columns. We're now turning it into a matrix (for plotting, analysis)
cor_df <- mi_corr %>%
  dplyr::select(variable1, variable2, r) %>%
  tidyr::pivot_wider(names_from = variable2, values_from = r)

# Rearranging columns and making diagonals 1 instead of NA
cor_df <- cor_df %>%
  dplyr::relocate(electricity_access, .before = income_growth)  %>%
  dplyr::mutate_all(~replace(., is.na(.), 1))

# Lastly, move var names from col 1 into row names
cor_df <- as.data.frame(cor_df)
rownames(cor_df) <- cor_df$variable1
cor_df <- cor_df[,-1]

#######
#### Figure 3: Corrrelation Plot
#######

# Set names for corplot printing
colnames(cor_df) <-  c("Electricity access", "Income growth",
                       "Income growth percent", "Children out of school",
                       "Infant mortality rate", "Primary completion rate",
                       "Real interest rate", "Percent pop. growth",
                       "Pop. density", "Total pop.",
                       "Health spending per capita",
                       "Health spending perc. GDP",
                       "Unemployment rate", "GDP growth",
                       "GDP per capita", "Birth rate",
                       "Adults HIV perc.", "Safe drinking water",
                       "Education duration")
```

```
581 rownames(cor_df) <- colnames(cor_df)
582
583 # and plot
584 corrplot(as.matrix(cor_df),tl.cex = 0.8, tl.col="black")
585
586
587 # Variance inflation factors (VIF). Find mean over imputations initially.
588 # Since we didn't learn a clear way to calculate VIF on multiply imputed
    datasets (and don't
589 # see one in 'miceadds'),
590 # we instead calculate it on each one individually and then report the mean
    VIF for each variable
591 # across the 10 datasets
592 vif_output <- list()
593 for (i in 1:10)
594 {
595   vif_output[[i]] <- faraway::vif(complete(imputations, i)[,-c(1,2)])
596 }
597
598 vif_output_means <- c()
599 for (i in 1:length(vif_output[[1]]))
600 {
601   val_vec <- c()
602   for (j in 1:10)
603   {
604     val_vec[j] <- vif_output[[j]][i]
605   }
606   vif_output_means[i] <- mean(val_vec)
607   names(vif_output_means)[i] <- names(vif_output[[1]][i])
608 }
609
610 # VIF. Find mean over imputations after
611 vif_output_means
612
613 # These are the variables with large VIFs and high pairwise correlations w/
    other vars
614 drop_vars <- c("Continent","life_expectancy","income_growth","birth_rate", "
    electricity_access", "health_expend_pc")
615
616 # Now recalculate VIFs after dropping the drop_vars
617 vif_output_after <- list()
618 for (i in 1:10)
619 {
```

```
  vif_output_after[[i]] <- faraway::vif(complete(imputations, i)[,!(names(
    complete(imputations, i)) %in% drop_vars)])
}

vif_output_means_after <- c()
for (i in 1:length(vif_output_after[[1]]))
{
  val_vec_after <- c()
  for (j in 1:10)
  {
    val_vec_after[j] <- vif_output_after[[j]][i]
  }
  vif_output_means_after[i] <- mean(val_vec_after)
  names(vif_output_means_after)[i] <- names(vif_output_after[[1]][i])
}

# Now we see lower means
vif_output_means_after

######
## Table S1
#######

# Somehow this table isn't printing the VIF after, but we manually added the
    values to the table
# from the above "vif_output_means_after"!

temp <- data.frame("Feature" = names(vif_output_means_after),
                   "VIF After Dropping Vars" = vif_output_means_after)

tableS1_df <- data.frame("Feature" = colnames(cor_df),
                         "VIF" = sprintf("%.2f",vif_output_means))
tableS1_df <- dplyr::left_join(tableS1_df, temp, by = "Feature")%>%
  dplyr::mutate(VIF.After.Dropping.Vars =
                  ifelse(is.na(VIF.After.Dropping.Vars),
                         " ",
                         sprintf("%.2f", VIF.After.Dropping.Vars)))

names(tableS1_df)[3] <- "VIF After Dropping Vars"


tableS1_table <- flextable(tableS1_df)

tableS1_table <- align(tableS1_table,j =1, align = "left", part = "all")
```

```
tableS1_table <- align(tableS1_table,j =2:3, align = "right", part = "all")
tableS1_table <- width(tableS1_table, width = c(1.4,1.0, 1.0))
#table1_table <- height_all(table1_table,height = 0.30)
tableS1_table <- hrule(tableS1_table, rule = "at least", part = "all")
tableS1_table # Note: we had to edit this in word after to add correct "
    after" values

######
### Question 3: Making the Model --------------
######

#Defining the analysis we're performing. Only include subset of variables (
    without drop_vars)
feature.selection1 <- expression(null.model1 <- lm(life_expectancy ~ 1),
    model2 <- step(null.model1, scope = ~  income_growth_pc + children_out_of
    _school + infant_mortality_rate + primary_completion_rate + real_interest
    _rate + pop_growth_perc_annual + pop_dens + pop_total + health_expend_
    perc_gdp + unemployment_rate + gdp_growth + gdp_pc + adults_HIV + safe_
    drinking_water + ed_duration, direction = "forward"))

# Redefine list of imputations to remove life expectancy for countries
# with NA life expectancy originally. We don't want to use them here - only
    in predictions later
countries_with_LE <- dplyr::filter(LE_data,
                                   !(is.na(life_expectancy)))$Country.Name


imp_filt <- mice::filter(imputations,
                         Country.Name %in% countries_with_LE)

# Using the with function to evaluate the above expression
step.fit <- with(imputations, feature.selection1)

# Extracting the number of times each variable appears in the m imputed
    datasets
step.fit <- with(imp_filt, feature.selection1)
step.fit.models <- lapply(step.fit$analyses, formula)
step.fit.features <- lapply(step.fit.models, terms)
feature.frequency <- unlist(lapply(step.fit.features, labels))

#Stepwise feature selection with our imputed datasets yields the following
    table
#showing the frequency of feature selection:
sort(table(feature.frequency), decreasing=TRUE)
```

```r

# Pool results according to above model (using mice, like we did in lab)
fit <- with(imp_filt, lm(life_expectancy ~
                           health_expend_perc_gdp + infant_mortality_rate +
    safe_drinking_water + pop_dens +
                           gdp_pc))

fit$analyses %>% str(max.level = 1)

pooled_results <- mice::pool(fit)
summary(pooled_results)
pool.r.squared(pooled_results, adjusted = TRUE)

#####
### Table 3: The final model
######

m <- 10
df <- as.data.frame(summary(pooled_results))
table3_df <- c()
table3_df$Feature <- c("Intercept",
                       "Health Expenditure (% GDP)",
                       "Infant mortality rate",
                       "% drinking safe water",
                       "Population density (per sq. km)",
                       "log(GDP per capita)")
table3_df$Estimate <- sprintf("%.4f", df$estimate)
table3_df$SE <- sprintf("%.5f", df$std.error)
table3_df$df <- sprintf("%.2f", df$df)
table3_df$P_val <- sprintf("%.4f", df$p.value)
table3_df$P_val <- ifelse(table3_df$P_val == "0.0000",
                          "< 0.0001", table3_df$P_val)
table3_df$BestSubsetFrequency <- c(" ",
                                   table(feature.frequency)[names(table(
    feature.frequency)) == "health_expend_perc_gdp"]/m,
                                   table(feature.frequency)[names(table(
    feature.frequency)) == "infant_mortality_rate"]/m,
                                   table(feature.frequency)[names(table(
    feature.frequency)) == "safe_drinking_water"]/m,
                                   table(feature.frequency)[names(table(
    feature.frequency)) == "pop_dens"]/m,
                                   table(feature.frequency)[names(table(
    feature.frequency)) == "gdp_pc"]/m
)
```

```r

# recalculate VIF for these five vars only
vif_output <- list()
for (i in 1:10){
  vif_output[[i]] <- faraway::vif(dplyr::select(
    complete(imputations, i), health_expend_perc_gdp,
    infant_mortality_rate,
    safe_drinking_water, pop_dens, gdp_pc))
}
vif_output_means2 <- c()
for (i in 1:length(vif_output[[1]])){
  val_vec <- c()
  for (j in 1:10){
    val_vec[j] <- vif_output[[j]][i]
  }
  vif_output_means2[i] <- mean(val_vec)
  names(vif_output_means2)[i] <- names(vif_output[[1]][i])
}
vif_output_means2
table3_df$VIF <- c(" ",vif_output_means2)
table3_df <- as.data.frame(table3_df)


names(table3_df) <- c("Feature",
                      "Estimate",
                      "Standard Error",
                      "df",
                      "P-value",
                      "Proportion of Best Subsets Appeared in",
                      "VIF")

table3_table <- flextable(table3_df, col_keys = c("Feature",
                                                  "Estimate",
                                                  "Standard Error",
                                                  "df",
                                                  "P-value",
                                                  "Proportion of Best
    Subsets Appeared in",
                                                  "VIF"))

table3_table <- align(table3_table,j =1, align = "left", part = "all")
table3_table <- align(table3_table,j =2:7, align = "right", part = "all")
table3_table <- width(table3_table, width = c(1.4,1.0, 1.1, 1.3, 1.0, 1.0,
    1.0 ))
```

```r
table3_table <- hrule(table3_table, rule = "at least", part = "all")
table3_table <- vline(table3_table, i = NULL, j = 5, part = "body",
                      border = fp_border(color="black"))
table3_table

#######
#### Figure 4: Residuals and model assumption check
#######


# Extract residuals to check model assumptions.
# We will do this for each imputed dataset individually and report averages.
# Should be ok with m = 10. A little higher and we might start to worry about CLT.
predict_df <- cbind(
  predict(fit$analyses[[1]], newdata = complete(imp_filt, 1)),
  predict(fit$analyses[[2]], newdata = complete(imp_filt, 2)),
  predict(fit$analyses[[3]], newdata = complete(imp_filt, 3)),
  predict(fit$analyses[[4]], newdata = complete(imp_filt, 4)),
  predict(fit$analyses[[5]], newdata = complete(imp_filt, 5)),
  predict(fit$analyses[[6]], newdata = complete(imp_filt, 6)),
  predict(fit$analyses[[7]], newdata = complete(imp_filt, 7)),
  predict(fit$analyses[[8]], newdata = complete(imp_filt, 8)),
  predict(fit$analyses[[9]], newdata = complete(imp_filt, 9)),
  predict(fit$analyses[[10]], newdata = complete(imp_filt, 10)))

observed_df <- cbind(
  dplyr::select(complete(imp_filt, 1), life_expectancy),
  dplyr::select(complete(imp_filt, 2), life_expectancy),
  dplyr::select(complete(imp_filt, 3), life_expectancy),
  dplyr::select(complete(imp_filt, 4), life_expectancy),
  dplyr::select(complete(imp_filt, 5), life_expectancy),
  dplyr::select(complete(imp_filt, 6), life_expectancy),
  dplyr::select(complete(imp_filt, 7), life_expectancy),
  dplyr::select(complete(imp_filt, 8), life_expectancy),
  dplyr::select(complete(imp_filt, 9), life_expectancy),
  dplyr::select(complete(imp_filt, 10), life_expectancy))

residual_df <- predict_df - observed_df # Note that residual_df has 10 columns (one for each imp) and 198 rows (one row per obs).
names(residual_df) <- c(1:10)


```

```r
# Now we find the mean for each obs and use that for residual diagnostic plots.
residual_means <- residual_df %>%
  dplyr::mutate(mean =rowMeans(residual_df))
residual_means <- residual_means[,"mean"]

# And do same for fit (for residual vs fit)
predicted_means <- as.data.frame(predict_df) %>%
  dplyr::mutate(mean = rowMeans(predict_df))
predicted_means <- predicted_means[,"mean"]

mean_df <- as.data.frame(cbind(predicted_means, residual_means))


# Plot

p1 <- ggplot(mean_df, aes(x=residual_means))+
  geom_histogram() + ggsci::scale_colour_jama() +
  theme_bw() + theme(plot.margin = unit(c(1,0.5,0.5,0.5), "cm")) +
  xlab("Residuals") +
  ylab("Count")

p2 <- ggplot(mean_df, aes(sample = residual_means)) +
  stat_qq() + stat_qq_line() + theme_bw()+
  theme(plot.margin = unit(c(1,0.5,0.5,0.5), "cm")) +
  xlab("Theoretical") +
  ylab("Sample")

p3 <- ggplot(mean_df, aes(x=predicted_means, y = residual_means))+
  geom_point() + theme_bw() +
  theme(plot.margin = unit(c(1,0.5,0.5,0.5), "cm")) +
  xlab("Fitted Value")+
  ylab("Residual")

# Plot's titles and format
top <- plot_grid(p1, p2, labels = c("A. Histogram of Residuals", "B. QQplot of Residuals"),
                 hjust = -0.1)
bottom <- plot_grid(p3, labels = "C. Residuals vs. Fit", hjust = -0.1)
plot_grid(top, bottom, nrow = 2)


######
### Table S2: Prediction for other 19 countries
```

```
######

# Generate predictions for the 19 countries initially missing LE and print
    results in table

# First re-add Country Name variable to each imputed dataset
datalist <- list(cbind(LE_data$Country.Name, complete(imputations,1)),
                 cbind(LE_data$Country.Name, complete(imputations,2)),
                 cbind(LE_data$Country.Name, complete(imputations,3)),
                 cbind(LE_data$Country.Name, complete(imputations,4)),
                 cbind(LE_data$Country.Name, complete(imputations,5)),
                 cbind(LE_data$Country.Name, complete(imputations,6)),
                 cbind(LE_data$Country.Name, complete(imputations,7)),
                 cbind(LE_data$Country.Name, complete(imputations,8)),
                 cbind(LE_data$Country.Name, complete(imputations,9)),
                 cbind(LE_data$Country.Name, complete(imputations,10)))

# And filter to only keep
datalist2 <- list()
for (i in 1:length(datalist)){
  names(datalist[[i]])[1] <- "Country.Name"
  datalist2[[i]] <- dplyr::filter(datalist[[i]],
                                  Country.Name %in% countries_missing_LE)
}

# Now predict over each one using coefficients from pooled model
# We aren't aware of an easy way to use 'predict()' on a model output by
    mice,
# so we perform predictions for estimates manually using the pooled
    coefficient estimates.
coefs <- pooled_results$pooled$estimate
pred_list <- list()

for (i in 1:length(datalist2)){
  df <- datalist2[[i]]
  pred_list[[i]] <- coefs[1] + coefs[2]*df$health_expend_perc_gdp +
    coefs[3]*df$infant_mortality_rate + coefs[4]*df$safe_drinking_water+
    coefs[5]*df$pop_dens + coefs[6]*df$gdp_pc
}

# Take mean of prediction for each countrry across imputed datasets
final_preds <- cbind(datalist2[[1]]$Country.Name, bind_cols(pred_list))

names(final_preds) = c("Country.Name",c("A1","A2",
```

```
                                          "A3", "A4",
                                          "A5", "A6",
                                          "A7","A8",
                                          "A9", "A10"))

final_preds <- dplyr::mutate(final_preds,
                             mean_pred =
                               rowMeans(dplyr::select(final_preds, starts_
    with("A"))))

# Now plot
tableS2_df <- dplyr::select(final_preds, Country.Name, mean_pred) %>%
  dplyr::mutate(mean_pred = sprintf("%.2f", mean_pred))

names(tableS2_df) <- c("Country", "Predicted life expectancy")


tableS2_table <- flextable(tableS2_df,
                           col_keys = c("Country",
                                        "Predicted life expectancy"))

tableS2_table <- align(tableS2_table,j =1, align = "left", part = "all")
tableS2_table <- align(tableS2_table,j =2, align = "right", part = "all")
tableS2_table <- width(tableS2_table, width = c(1.4,1.0))
#table1_table <- height_all(table1_table,height = 0.30)
tableS2_table <- hrule(tableS2_table, rule = "at least", part = "all")
tableS2_table

######
### Question 4: Experimental Design ------------------------
######

# Performing ANOVA III
anova_LE <- aov(life_expectancy ~ as.factor(Continent), data=complete(
    imputations))
summary(anova_LE)
Anova(anova_LE, type = "III")

# This specific section of code wasn't from lab.
# We used this R-bloggers post for reference [5] here:
# https://www.r-bloggers.com/2021/07/how-to-perform-ancova-in-r/

# We just pick complete(imputations, 1) as our single imputation
anova_data <- complete(imputations, 1)
```

```r
anova_data$Continent <- as.factor(anova_data$Continent)

# Performing ANCOVA (according to slides from Lecture 10)
le.aov <- aov(life_expectancy ~ log(gdp_pc) + Continent, data = anova_data)
summary(le.aov)

# Post-hoc test
tukey.le <- glht(le.aov, linfct = mcp(Continent = "Tukey"))

# view a summary of the post-hoc comparisons
summary(tukey.le)

# Note: we also trried re-running above code using complete(imputations, i)
# With i = 2,3,...,10 as sensitivity analysis and got similar results

#Before proceeding with the statistical tests, we check the assumptions of
#ANCOVA which are (show diagnostic plots in appendix):

#####
### Figure 5:
#####

# Plot life expectancy by % of GDP spent on healthcare
p1 <- ggplot(anova_data, aes(x = log(gdp_pc), y = life_expectancy, color =
    Continent)) +
  theme_bw() +
  geom_point(alpha = 0.5, size = 0.8) + xlab("log(GDP per capita in
    thousands)")+
  ylab("Life expectancy") + ggsci::scale_color_jama() + geom_smooth(method="
    lm",se = FALSE)

# Figure showing life expectancy by continent (with singly imputed dataset
    that we will use)
p2 <- ggplot(complete(imputations, 1), aes(x = Continent, y = life_
    expectancy, color = Continent)) +
  geom_boxplot(show.legend = FALSE) + ggsci::scale_color_jama() + theme_bw()
    + theme(axis.text.x = element_text(angle = -45),

            plot.margin = unit(c(1,0.5,0.5,0.5), "cm")) + xlab("Continent")
    + ylab("Life Expectancy (Years)")

plot_grid(p1, p2, labels = c("A", "B"))

# Test for homogeneity of variances
```

```
car::leveneTest(anova_data$life_expectancy~anova_data$Continent)

# Analysing residual's normality assumptions
anova_data$residuals <- le.aov$residuals


#####
### Figure S2
#####

p1 <- ggplot(anova_data, aes(residuals)) +
  geom_histogram(color = "#000000", fill = "#0099F8") +
  theme_bw()+
  xlab("Residual")+
  ylab("Frequency")+
  theme(plot.margin = unit(c(1,0.5,0.5,0.5), "cm"))

# Plot 2
p2 <- ggplot(anova_data, aes(sample = residuals)) +
  stat_qq() +
  stat_qq_line() +
  theme_bw()+
  theme(plot.margin = unit(c(1,0.5,0.5,0.5), "cm")) +
  xlab("Theoretical") +
  ylab("Sample")

plot_grid(p1, p2, labels = c("A.", "B"))

# Shapiro test
shapiro.test(anova_data$residuals)
```

8 Statement on contribution of group members

Students [2200634] and [2200909] participated equally in this project. We worked together in-person to create the bones of the code, write much of the report, and prepare the slides. For the parts that we did split up, [2200909] worked on the figures/tables and model fine-tuning while [2200634] worked on the continent model, documentation, and assembling the slides. We still reviewed and worked on the final version together for each of these tasks. One more student (unknown registration number) was also assigned to our group but did not contribute at all because they stopped responding to our messages or contacting us.