STA108 Term Project

Gabriel Jones

2023-12-08

All files related to this project can be accessed via https://github.com/gqjones/STA108_TermProject

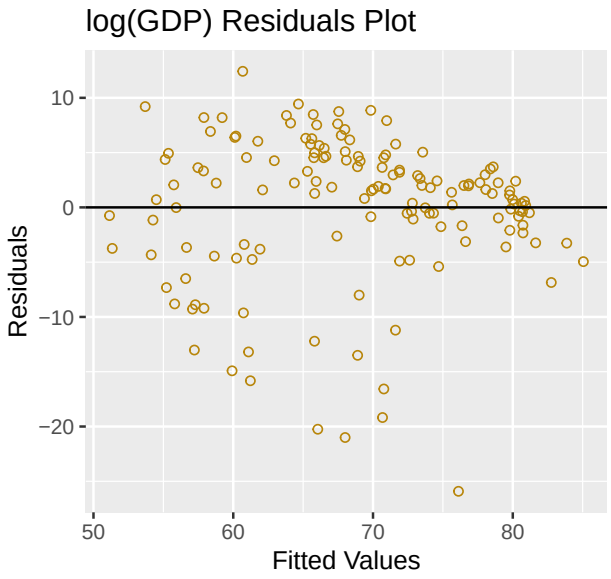
Introduction

In this project, we are interested in building a parsimonious model to estimate the life expectancy of individuals depending on various aspects of their residence country. The factors that will be considered for this model include: *Land Area*, *Population*, *Rural Population*, *Health Care Expenditures*, *Internet Access*, *Birth Rate*, *Elderly Population*, *CO2 Emissions*, *GDP*, and *Cell Phone Subscriptions*. The data set provided contains this information for 148 different countries.

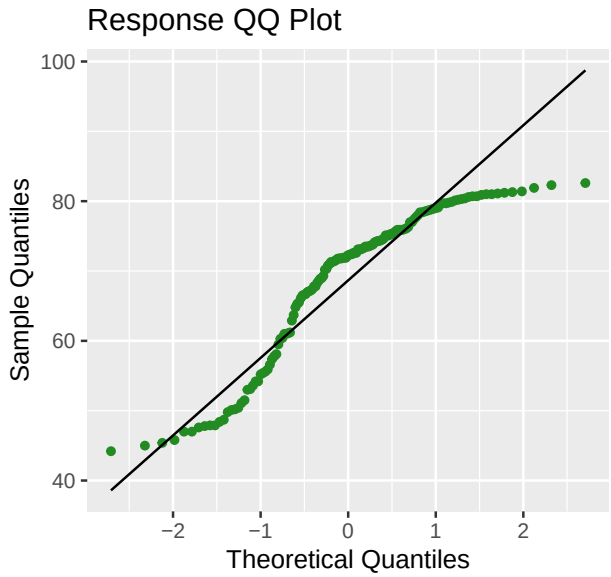
Methods

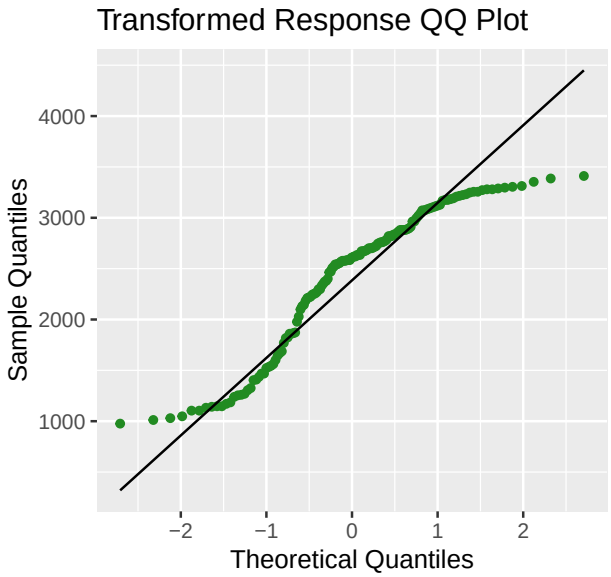
We began our analysis by fitting single factor linear models with each of our potential predictors on life expectancy using the `lm()` function in R. This provided us with estimates for the regression parameter, R^2 values, and t-test results to test if the predictor may not have any relationship with life expectancy. From this information, we found that the predictors *Land Area* and *Population* had no significant affect on Life Expectancy, thus they were no longer considered as predictors in our model.

We then plotted the relationships between our predictor and outcome variables and carried out a residual analysis for each model to determine any deviations from standard linear regression model assumptions. From this, we found that *CO2*, *Elderly Population*, *GDP*, and *Internet* had abnormal residual plots and non-linearity with life expectancy. To remedy this, we took the natural logarithm of each of these predictors. For the most part, this resolved the issues with these models with the exception of *Elderly Population* barely exhibiting any change from its previous diagnostics. Below shows the change in residual plots after the transformation. This same change was exhibited in *CO2* and *Internet*.



Additionally, we noticed that the residual plots for all the single factor linear models exhibited heteroscedasticity. This suggested that there may be an issue with our response variable *Life Expectancy*. After creating a QQ plot for life expectancy, we saw that the data was left skewed. As an attempt to remedy this, we performed a boxcox transformation on *Life Expectancy* using $\lambda = 2$, but found that there was minimal change in the normality of the data. After performing a Shapiro-Wilk normality test to ensure there was no change in normality, we decided not to use the boxcox transformed response variable. The graphs below show the lack of change in normality.





Before moving on to variable selection and building of our multifactor model, we assessed the potential for multicollinearity in each pair of our variables. We did this by finding the VIF values for each pair of variables, then displaying them together as a matrix.

Table 1: VIF Values

	LandArea	Population	Rural	Health	Internet	BirthRate	ElderlyPop	CO2	GDP	Cell
LandArea	10.000	1.260	1.022	1.002	1.009	1.010	1.016	1.023	1.011	1.002
Population	1.260	10.000	1.006	1.010	1.000	1.004	1.001	1.001	1.002	1.009
Rural	1.022	1.006	10.000	1.023	1.775	1.550	1.231	1.791	2.306	1.675
Health	1.002	1.010	1.023	10.000	1.084	1.062	1.161	1.008	1.093	1.009
Internet	1.009	1.000	1.775	1.084	10.000	2.672	1.750	2.659	3.283	1.999
BirthRate	1.010	1.004	1.550	1.062	2.672	10.000	2.549	2.973	2.799	1.823
ElderlyPop	1.016	1.001	1.231	1.161	1.750	2.549	10.000	1.401	1.627	1.270
CO2	1.023	1.001	1.791	1.008	2.659	2.973	1.401	10.000	4.304	2.087
GDP	1.011	1.002	2.306	1.093	3.283	2.799	1.627	4.304	10.000	2.175
Cell	1.002	1.009	1.675	1.009	1.999	1.823	1.270	2.087	2.175	10.000

From the above table, we can see that *CO2* and *GDP* have the most collinearity with each other. These two variables also exhibit VIF values over 2 for other variables as well. As such, we determined it may be beneficial to exclude these variables from our model.

To select variables for our model, we used both forward selection and backward elimination algorithms provided by the `LEAPS` package. We ran the two algorithms on the model excluding *Land Area* and *Population* and the natural log transformations of *CO2*, *Elderly Population*, *GDP*, and *Internet*. Both algorithms came to the same result, so we used the specified variables in our final model.

Results

From our analysis, the variables we found to be most accurate in predicting life expectancy were: *Rural Population*, *Health*, *Internet*, *Birth Rate*, and *Elderly Population*. We found that the variables *Internet* and *Elderly Population* exhibited a logistic relationship with life expectancy, while the other three variables exhibited a linear relationship. The final model had $R^2 = 0.785$. The *elderly population*, *birth rate*, and *% rural population* were shown to have a negative relationship with life expectancy, thus, minimizing these factors would work to maximize life expectancy. Below details our model found to best predict life expectancy along with explicit definitions for each predictor.

$X_1 =$ % Rural Population

$X_2 =$ % of $ For Healthcare

$X_3 =$ % of Population With Internet Access

$X_4 =$ Birth Rate per 1000 Individuals

$X_5 =$ % Elderly Population

$$LifeExpectancy = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 log(X_3) + \beta_4 X_4 + \beta_5 log(X_5) + \epsilon$$

Below details our estimates for $\beta_1, ..., \beta_5$

	b0	b1	b2	b3	b4	b5
Estimate	82.24408	-0.0460924	0.2307352	1.588339	-0.705433	-1.511908

Summary

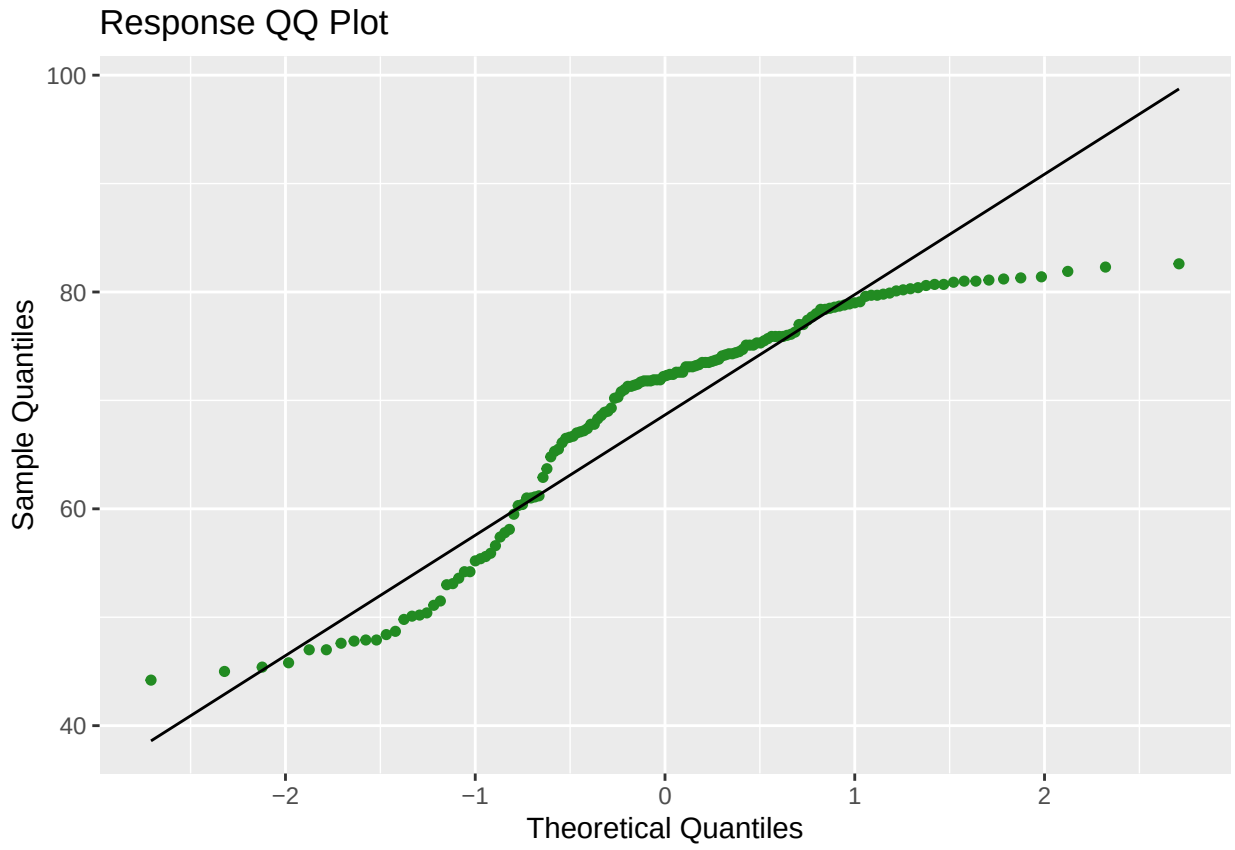
The goal of this project was to build a parsimonious model that estimates the life expectancy of an individual based on data from the country they live in. After carrying out residual analysis, variable transformations, and variable selection algorithms, we found the best variables used in predicting life expectancy were *Rural Population*, *Health*, *Internet*, *Birth Rate*, and *Elderly Population*. Our model does an adequate job at estimating life expectancy, however, there are some improvements that can increase the effectiveness of this model. One improvement would be to use a more effective transformation of the *Elderly Population* variable to resolve the non constant error. This model could also be improved by resolving the Heteroscedasticity of the residuals in our model.

Appendix

```
## Warning: package 'leaps' was built under R version 4.3.2
```

Single Linear Regression Models & Analysis

Life Expectancy Analysis

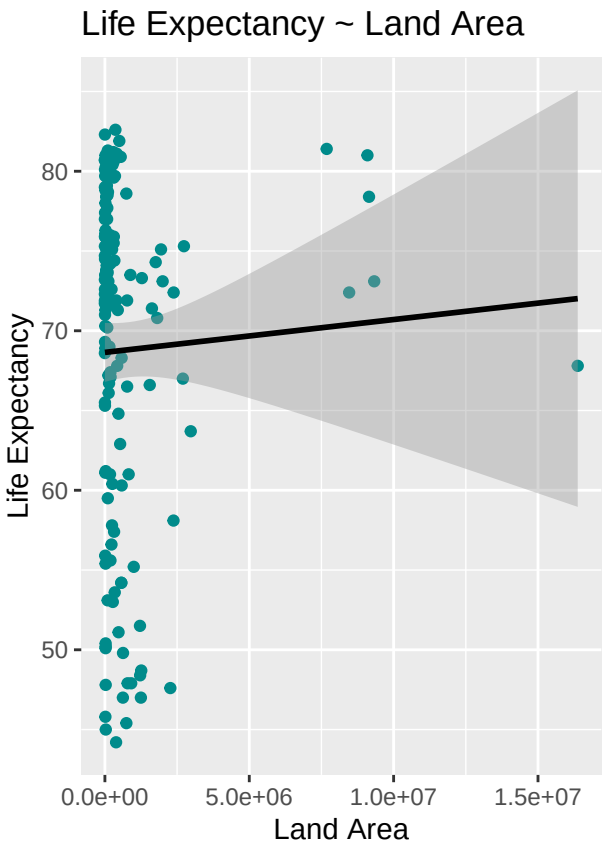


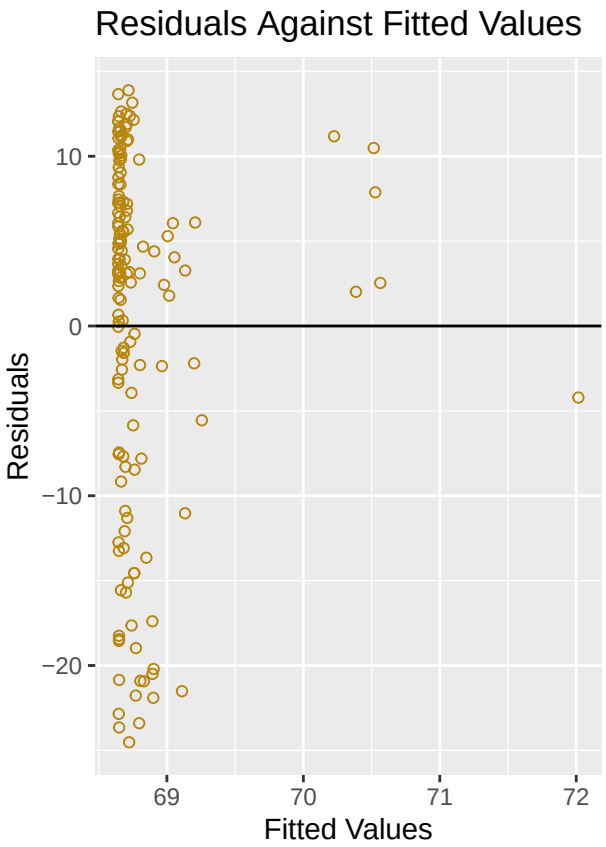
Life Expectancy ~ Land Area

Table 3: Life Expectancy ~ Land Area

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	68.6437149	0.9344772	73.4568137	0.0000000
Land Area	0.0000002	0.0000004	0.4899179	0.6249274

```
## Saving 6.5 x 4.5 in image
## `geom_smooth()` using formula = 'y ~ x'
## Saving 6.5 x 4.5 in image
## Saving 6.5 x 4.5 in image
## Saving 6.5 x 4.5 in image
## `geom_smooth()` using formula = 'y ~ x'
```


Life Expectancy ~ Land Area
Life Expectancy
80
70
60
50
0.0e+00
5.0e+06
1.0e+07
1.5e+07
Land Area

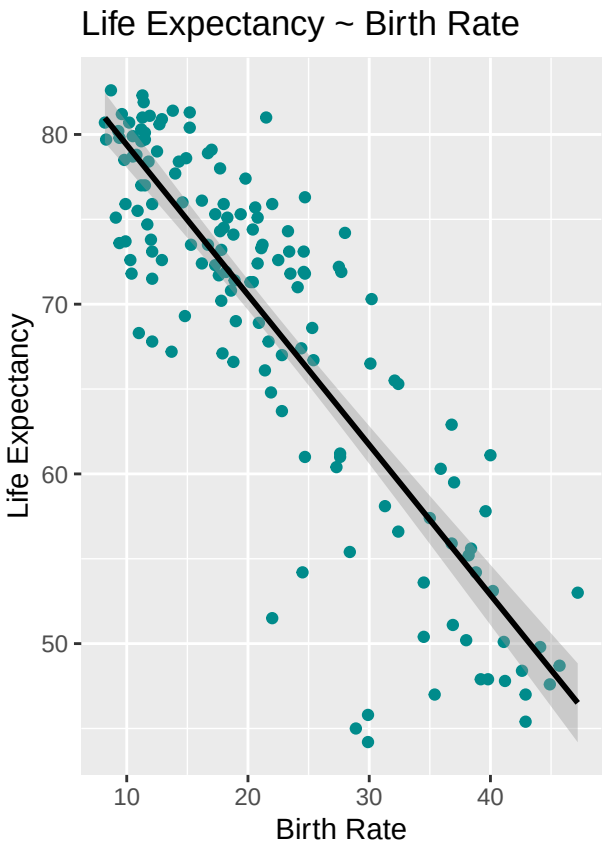

Residuals Against Fitted Values
Residuals
10
0
−10
−20
69
70
71
72
Fitted Values

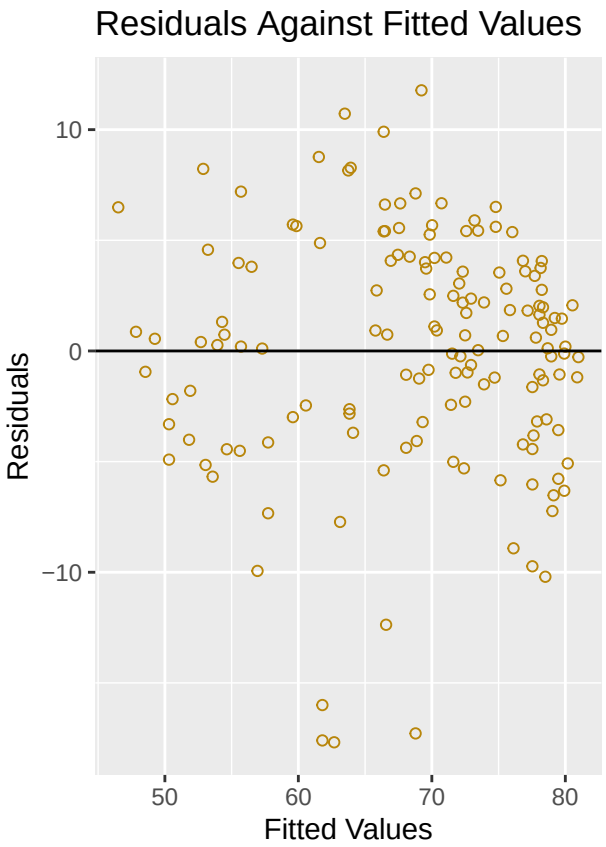
Life Expectancy ~ Birth Rate

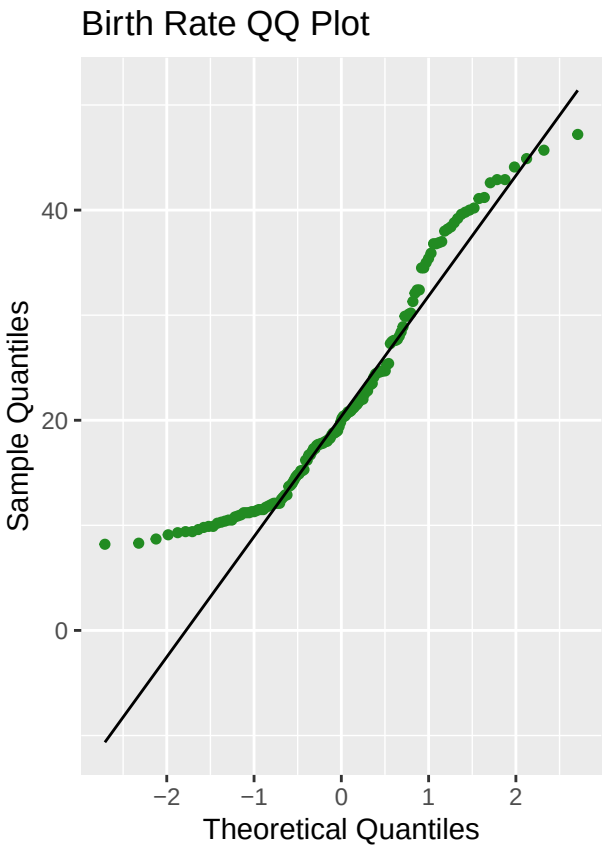
Table 4: Life Expectancy ~ Birth Rate

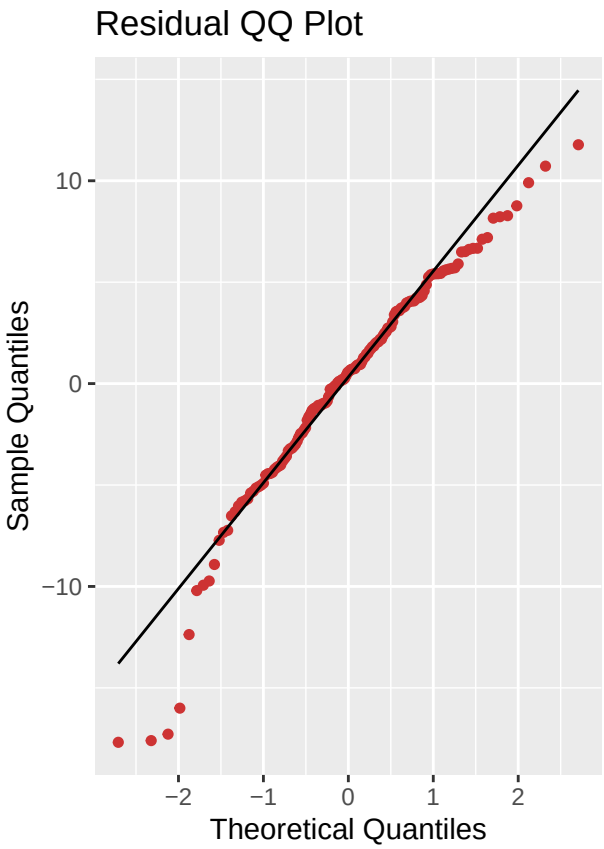
	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	88.2252805	1.0469920	84.26548	0
Birth Rate	-0.8837966	0.0431595	-20.47747	0

```
## Saving 6.5 x 4.5 in image
## `geom_smooth()` using formula = 'y ~ x'
## Saving 6.5 x 4.5 in image
## Saving 6.5 x 4.5 in image
## Saving 6.5 x 4.5 in image
## `geom_smooth()` using formula = 'y ~ x'
```


Life Expectancy ~ Birth Rate
Life Expectancy
80
70
60
50
10
20
30
40
Birth Rate


Residuals Against Fitted Values
Residuals
10
0
−10
50
60
70
80
Fitted Values



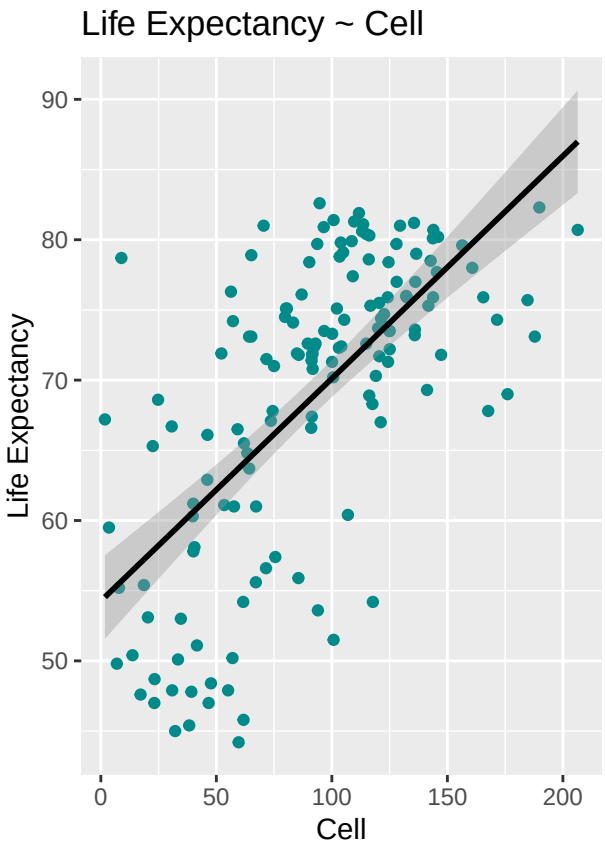


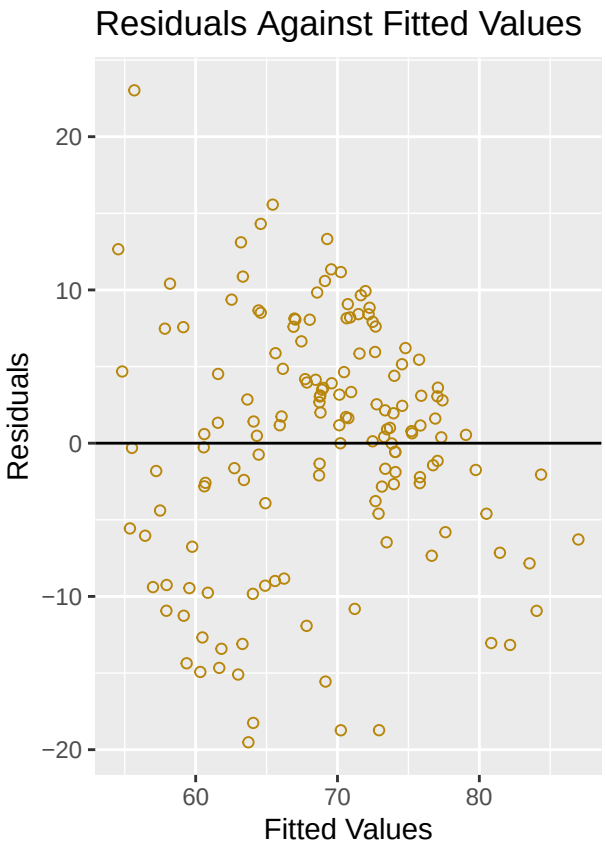
Life Expectancy ~ Cell

Table 5: Life Expectancy ~ Cell

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	54.262300	1.5330388	35.39526	0
Cell	0.158516	0.0150848	10.50833	0

```
## Saving 6.5 x 4.5 in image
## `geom_smooth()` using formula = 'y ~ x'
## Saving 6.5 x 4.5 in image
## Saving 6.5 x 4.5 in image
## Saving 6.5 x 4.5 in image
## `geom_smooth()` using formula = 'y ~ x'
```


Life Expectancy ~ Cell
Life Expectancy
Cell
90
80
70
60
50
0
50
100
150
200

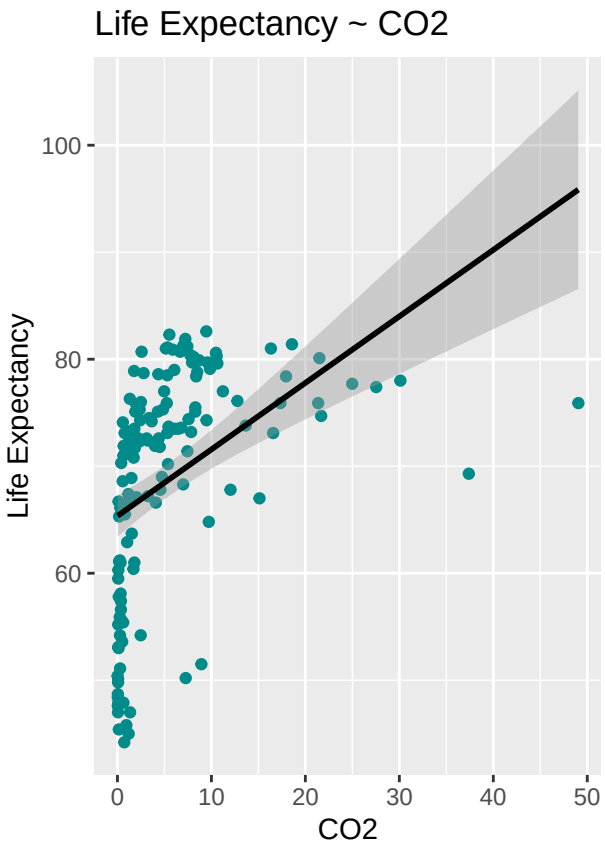

Residuals Against Fitted Values
Residuals
Fitted Values
20
10
0
−10
−20
60
70
80

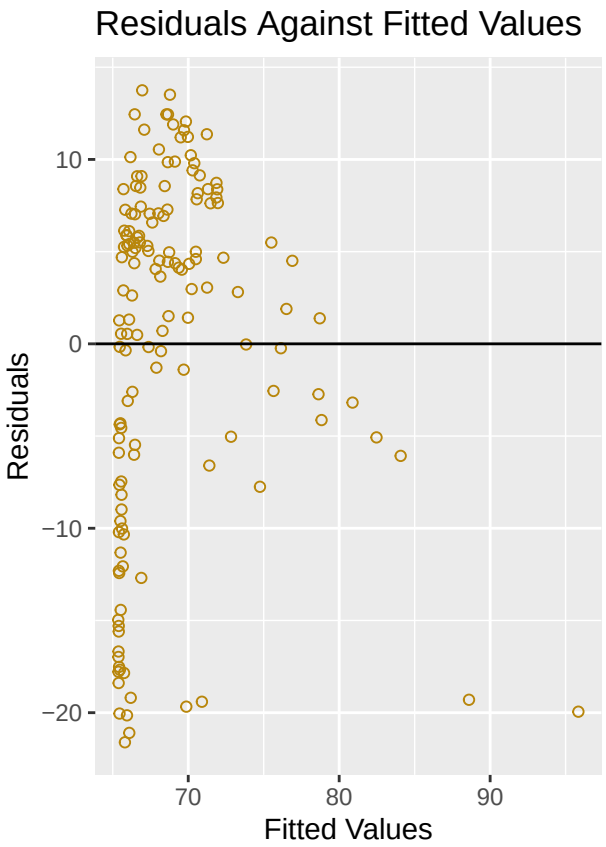
Life Expectancy ~ CO2

Table 6: Life Expectancy ~ CO2

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	65.3529718	0.9840266	66.413824	0
CO2	0.6216755	0.1064559	5.839747	0

```
## Saving 6.5 x 4.5 in image
## `geom_smooth()` using formula = 'y ~ x'
## Saving 6.5 x 4.5 in image
## Saving 6.5 x 4.5 in image
## Saving 6.5 x 4.5 in image
## `geom_smooth()` using formula = 'y ~ x'
```


Life Expectancy ~ CO2
Life Expectancy
CO2
100
80
60
0
10
20
30
40
50

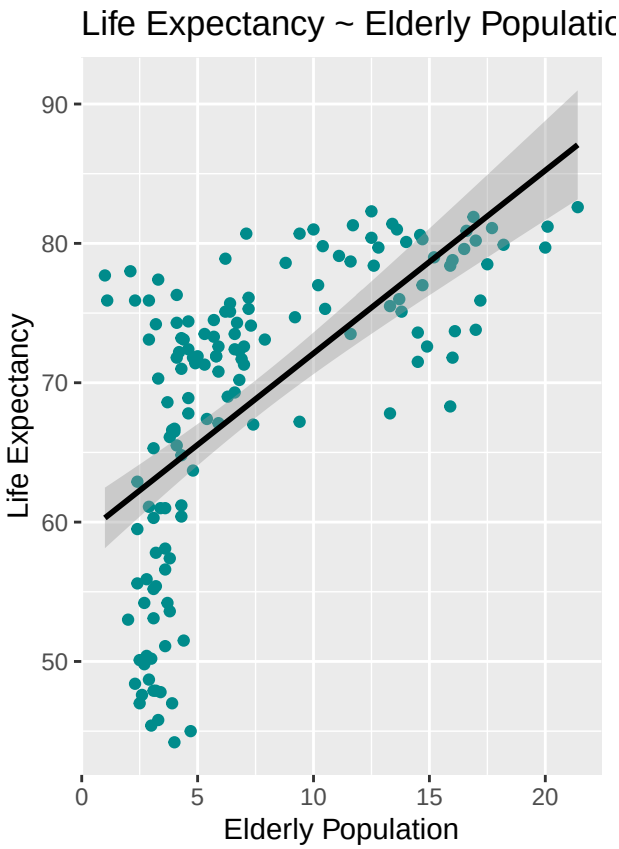

Residuals Against Fitted Values
Residuals
Fitted Values
10
0
−10
−20
70
80
90

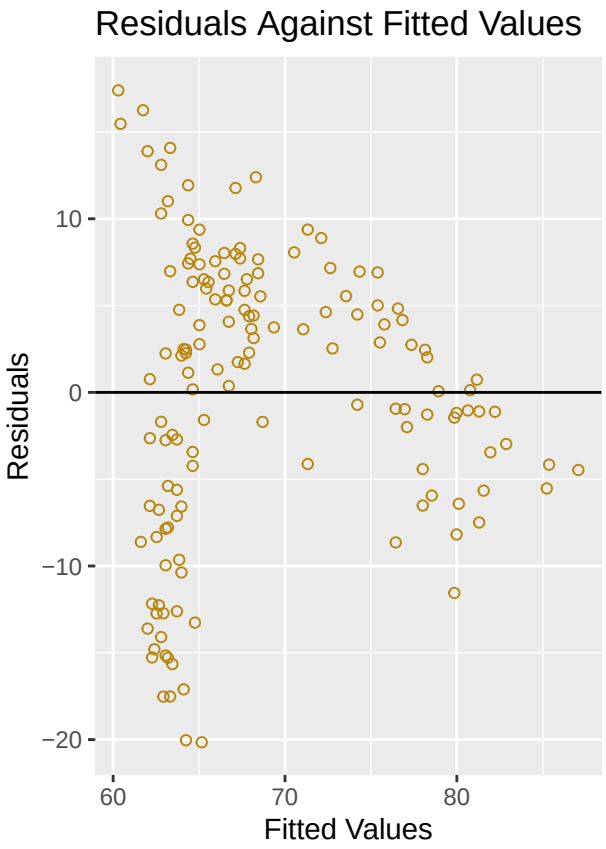
Life Expectancy ~ Elderly Population

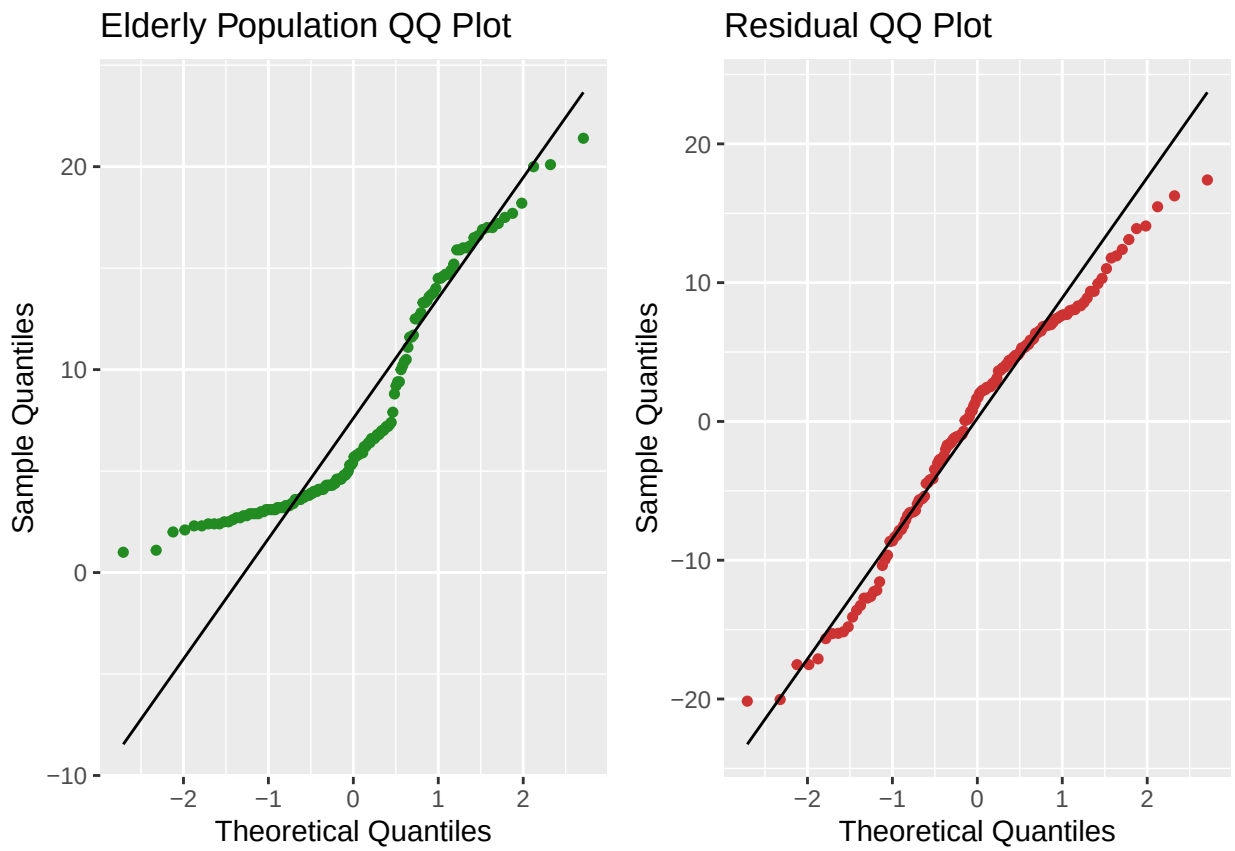
Table 7: Life Expectancy ~ Elderly Population

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	58.987283	1.2078486	48.836652	0
Elderly Population	1.312309	0.1336064	9.822199	0

```
## Saving 6.5 x 4.5 in image
## `geom_smooth()` using formula = 'y ~ x'
## Saving 6.5 x 4.5 in image
## Saving 6.5 x 4.5 in image
## Saving 6.5 x 4.5 in image
## `geom_smooth()` using formula = 'y ~ x'
```


Life Expectancy ~ Elderly Populatic
Life Expectancy
Elderly Population
90
80
70
60
50
0
5
10
15
20


Residuals Against Fitted Values
Residuals
Fitted Values
10
0
−10
−20
60
70
80

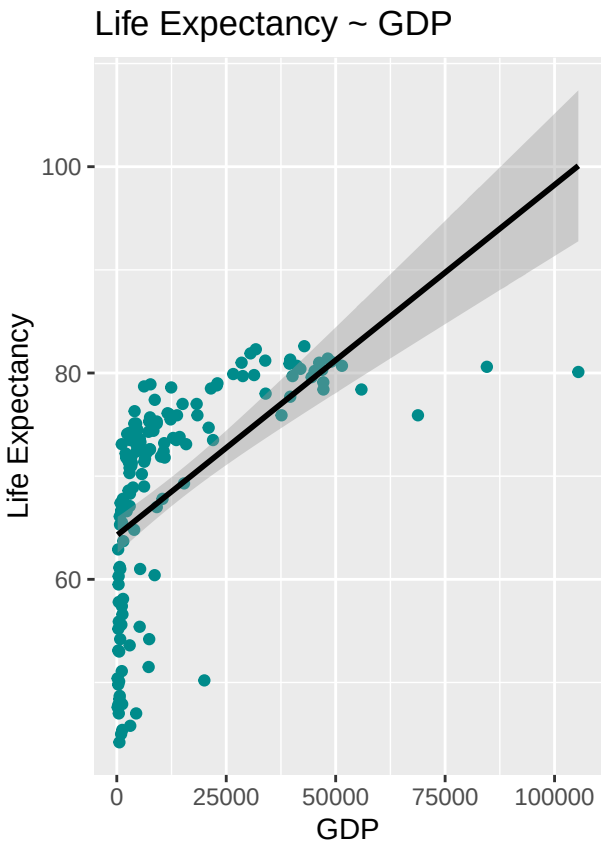


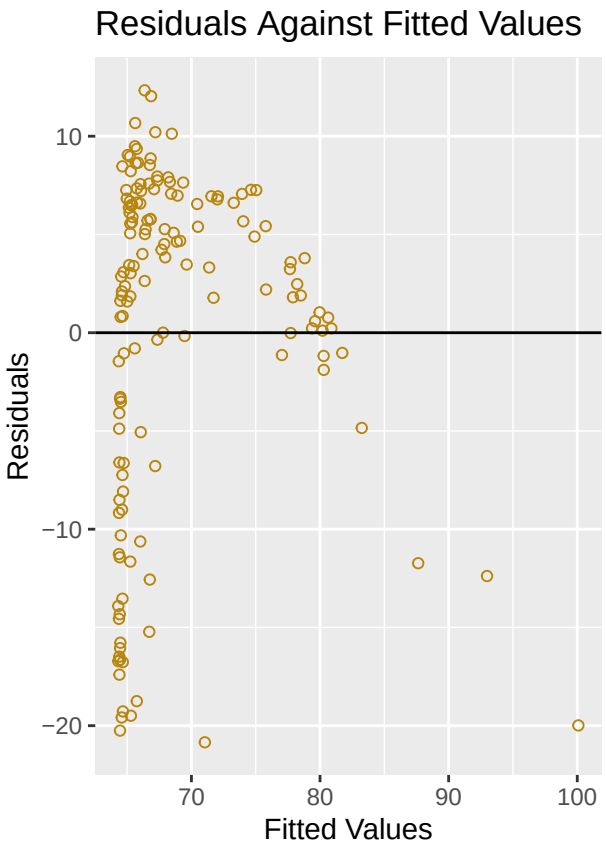
Life Expectancy ~ GDP

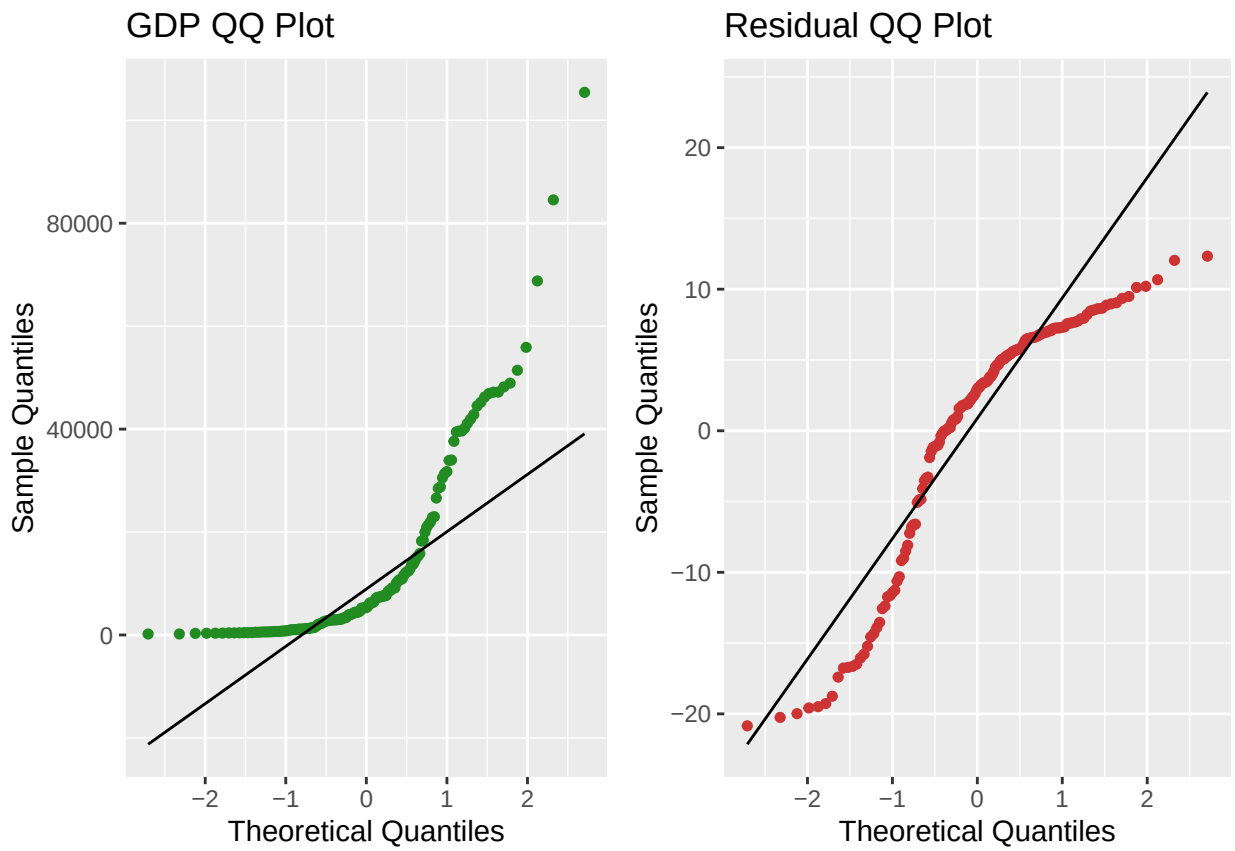
Table 8: Life Expectancy ~ GDP

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	64.2487516	0.8867131	72.457200	0
GDP	0.0003399	0.0000395	8.610075	0

```
## Saving 6.5 x 4.5 in image
## `geom_smooth()` using formula = 'y ~ x'
## Saving 6.5 x 4.5 in image
## Saving 6.5 x 4.5 in image
## Saving 6.5 x 4.5 in image
## `geom_smooth()` using formula = 'y ~ x'
```


Life Expectancy ~ GDP
Life Expectancy
100
80
60
0
25000
50000
75000
100000
GDP


Residuals Against Fitted Values
Residuals
10
0
−10
−20
70
80
90
100
Fitted Values

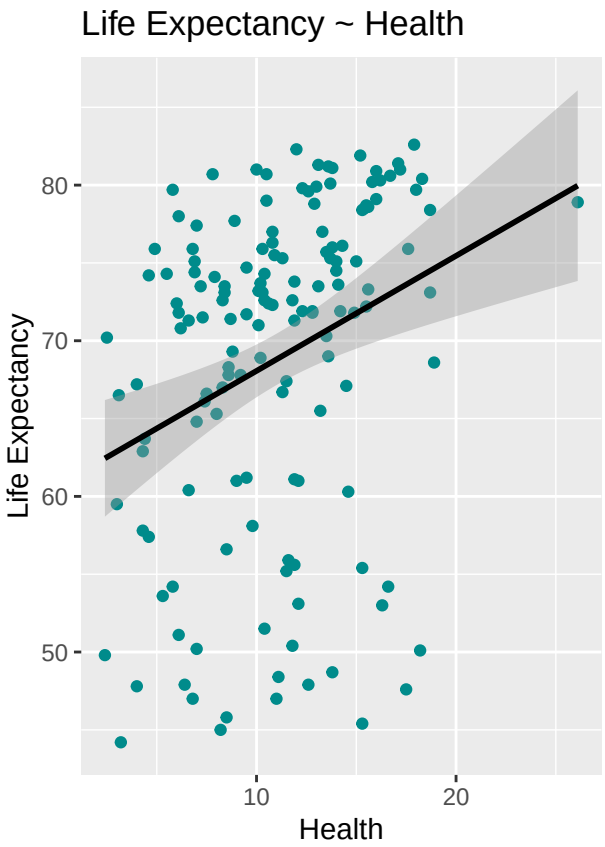


Life Expectancy ~ Health

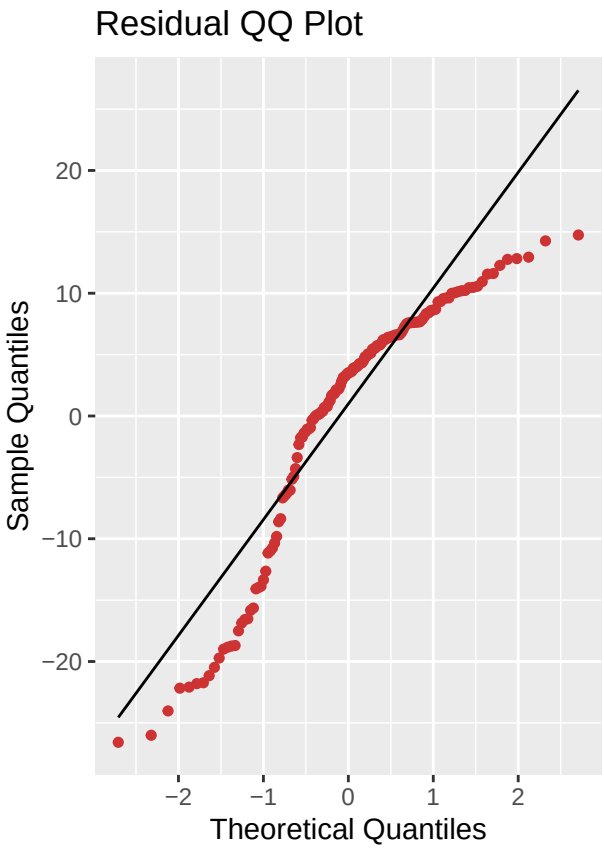
Table 9: Life Expectancy ~ Health

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	60.6696446	2.3307451	26.030150	0.0000000
Health	0.7392994	0.1977105	3.739303	0.0002644

```
## Saving 6.5 x 4.5 in image
## `geom_smooth()` using formula = 'y ~ x'
## Saving 6.5 x 4.5 in image
## Saving 6.5 x 4.5 in image
## Saving 6.5 x 4.5 in image
## `geom_smooth()` using formula = 'y ~ x'
```


Life Expectancy ~ Health
Life Expectancy
80
70
60
50
10
20
Health


Residuals Against Fitted Values
Residuals
10
0
−10
−20
65
70
75
80
Fitted Values

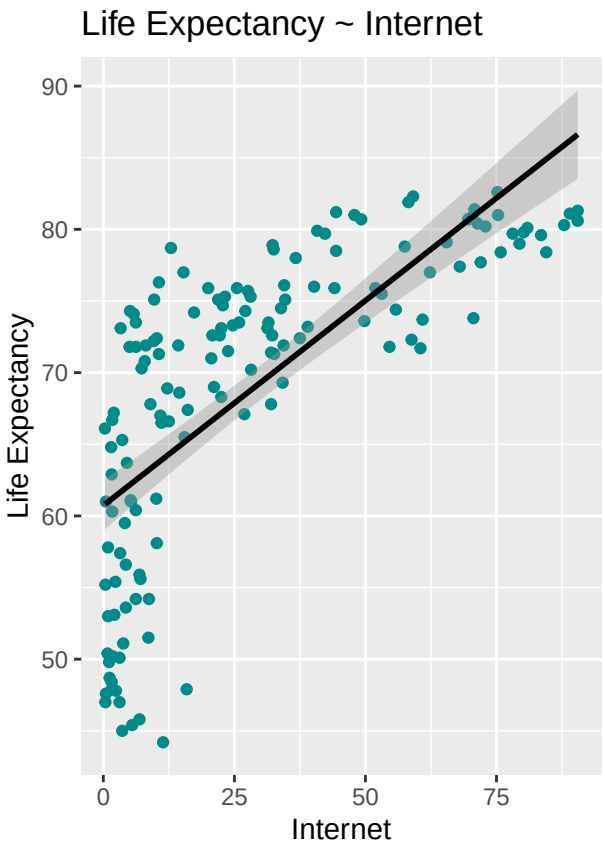


Life Expectancy ~ Internet

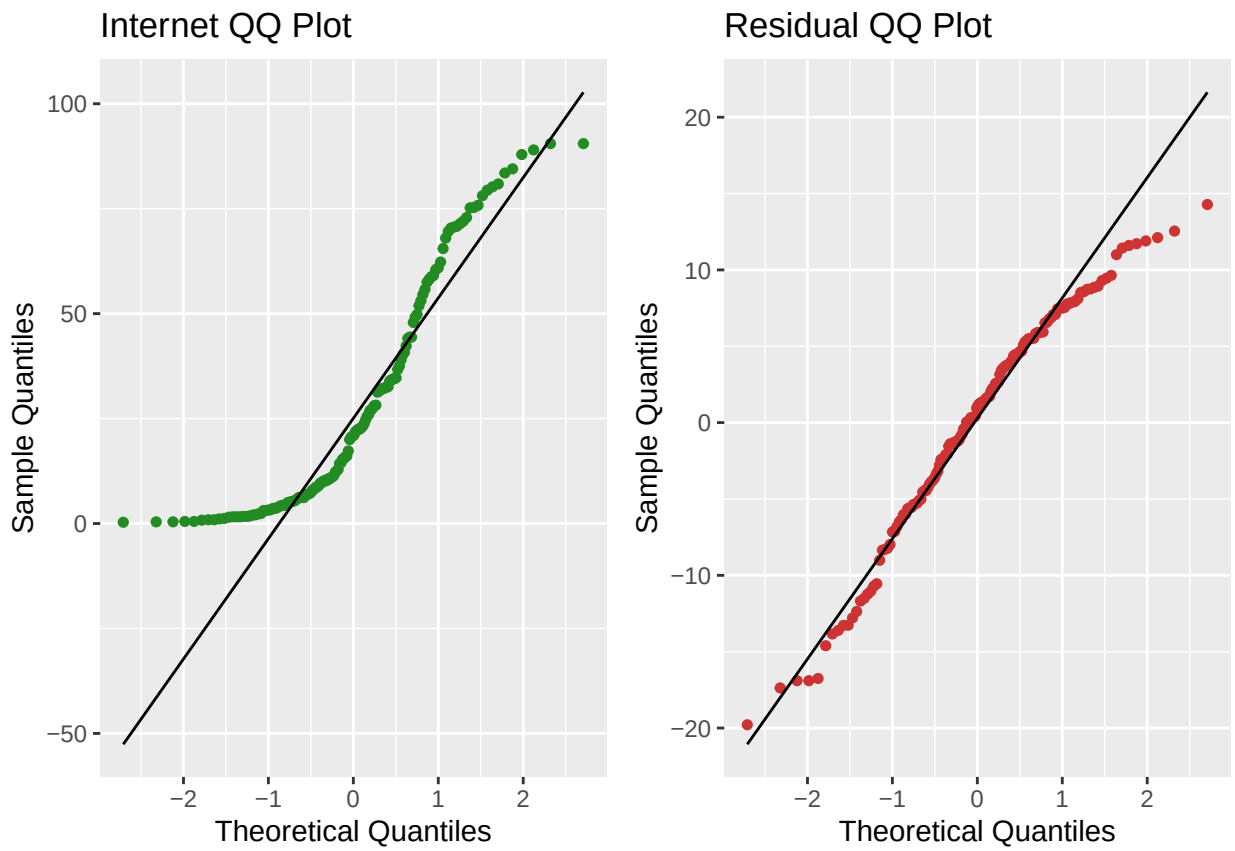
Table 10: Life Expectancy ~ Internet

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	60.725099	0.8927376	68.02122	0
Internet	0.286114	0.0230849	12.39400	0

```
## Saving 6.5 x 4.5 in image
## `geom_smooth()` using formula = 'y ~ x'
## Saving 6.5 x 4.5 in image
## Saving 6.5 x 4.5 in image
## Saving 6.5 x 4.5 in image
## `geom_smooth()` using formula = 'y ~ x'
```


Life Expectancy ~ Internet
Life Expectancy
Internet
90
80
70
60
50
0
25
50
75


Residuals Against Fitted Values
Residuals
Fitted Values
10
0
−10
−20
60
70
80

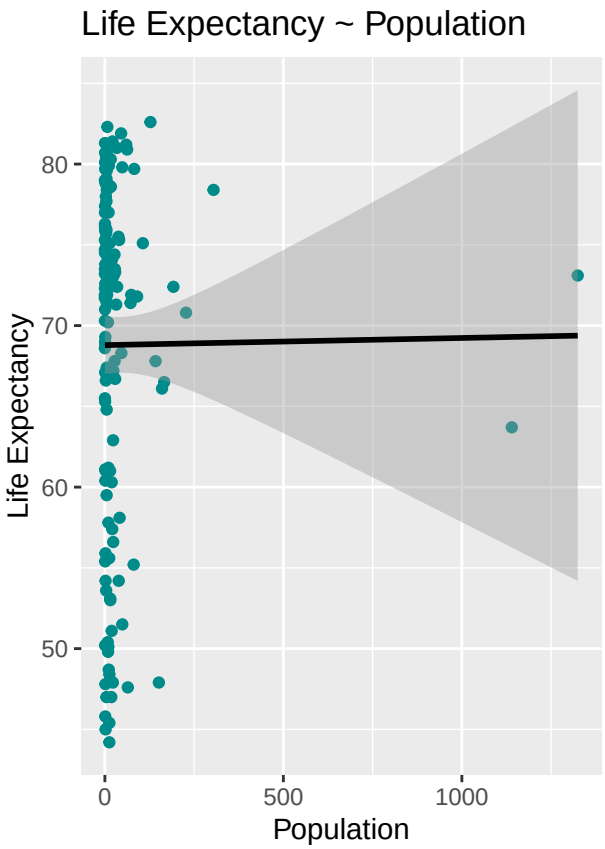


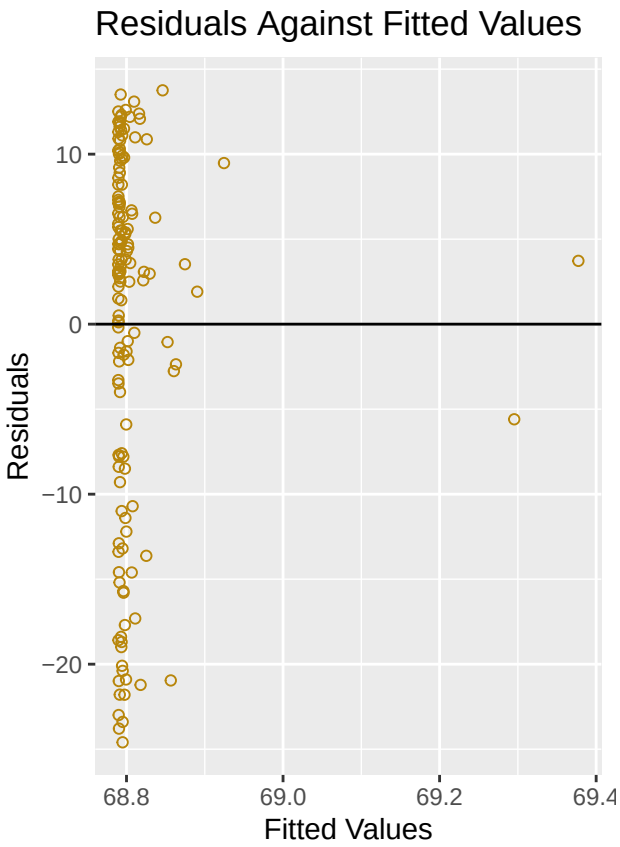
Life Expectancy ~ Population

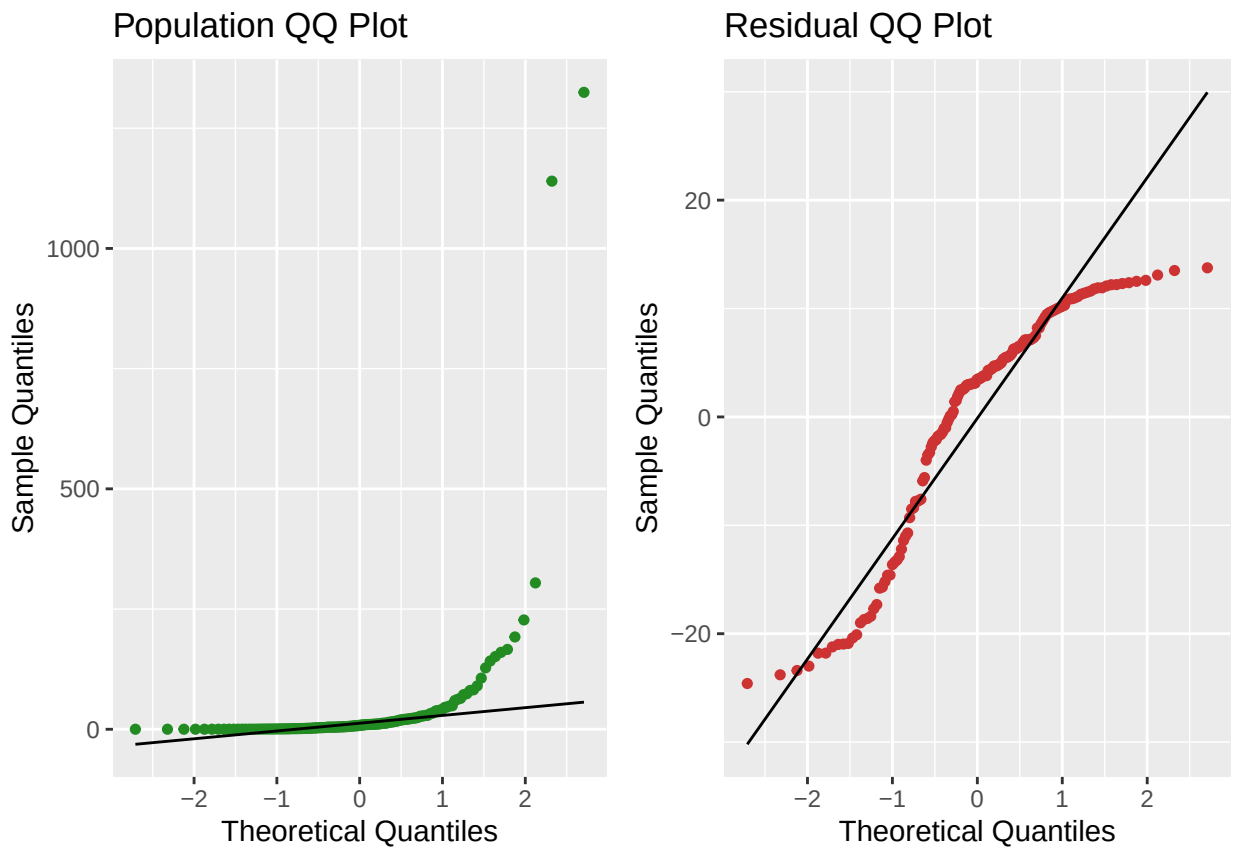
Table 11: Life Expectancy ~ Population

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	68.7894859	0.9058768	75.9369106	0.0000000
Population	0.0004438	0.0059454	0.0746386	0.9406045

```
## Saving 6.5 x 4.5 in image
## `geom_smooth()` using formula = 'y ~ x'
## Saving 6.5 x 4.5 in image
## Saving 6.5 x 4.5 in image
## Saving 6.5 x 4.5 in image
## `geom_smooth()` using formula = 'y ~ x'
```


Life Expectancy ~ Population
Life Expectancy
80
70
60
50
0
500
1000
Population


Residuals Against Fitted Values
Residuals
10
0
−10
−20
68.8
69.0
69.2
69.4
Fitted Values

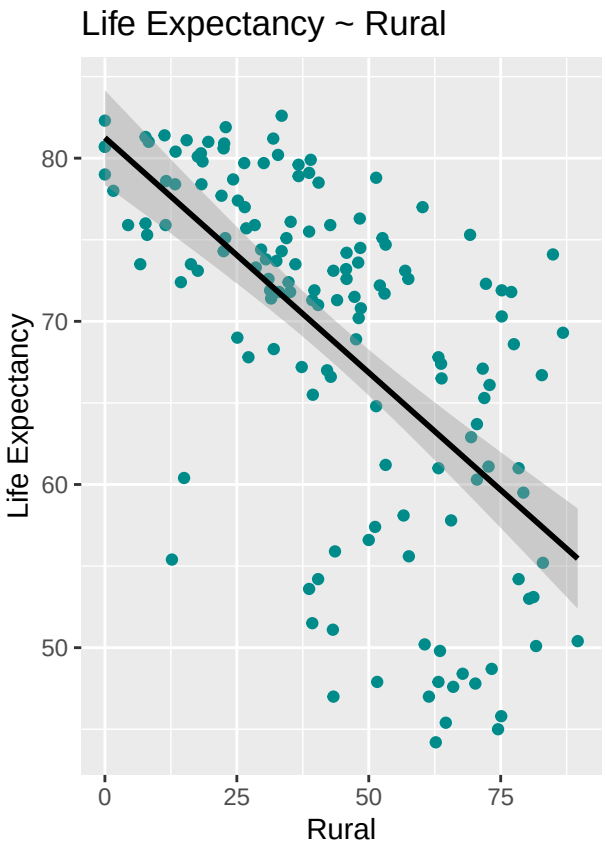


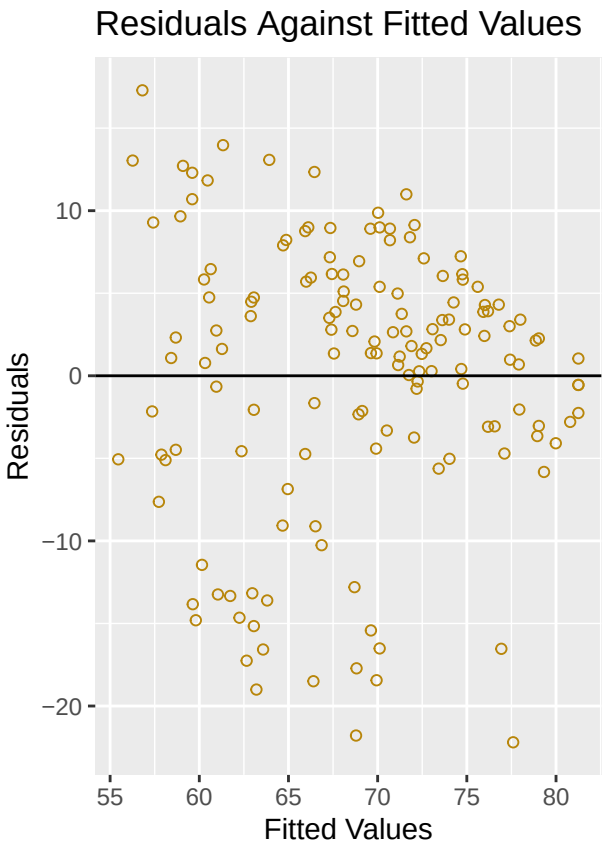
Life Expectancy ~ Rural

Table 12: Life Expectancy ~ Rural

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	81.2538126	1.4694641	55.294859	0
Rural	-0.2878945	0.0300757	-9.572321	0

```
## Saving 6.5 x 4.5 in image
## `geom_smooth()` using formula = 'y ~ x'
## Saving 6.5 x 4.5 in image
## Saving 6.5 x 4.5 in image
## Saving 6.5 x 4.5 in image
## `geom_smooth()` using formula = 'y ~ x'
```


Life Expectancy ~ Rural
Life Expectancy
80
70
60
50
0
25
50
75
Rural


Residuals Against Fitted Values
Residuals
10
0
−10
−20
55
60
65
70
75
80
Fitted Values

Transformations

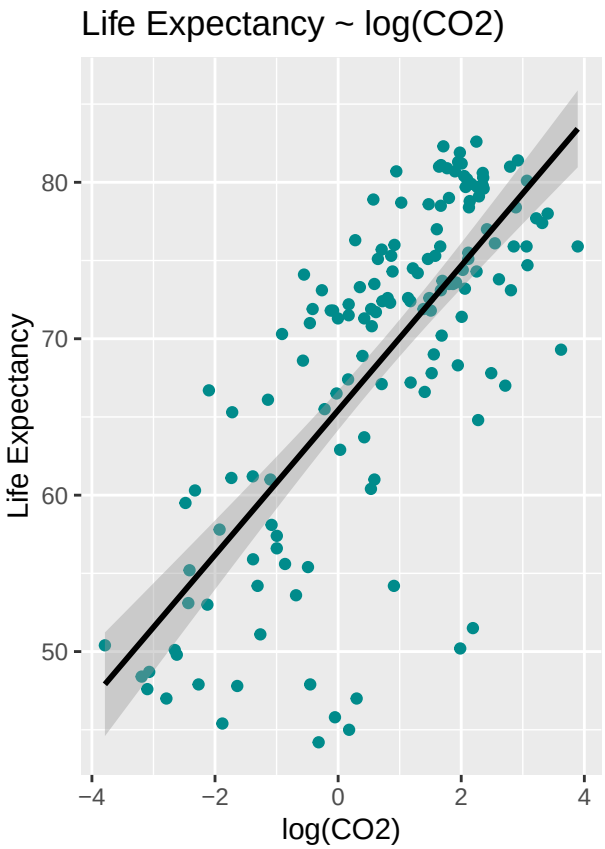
New Models

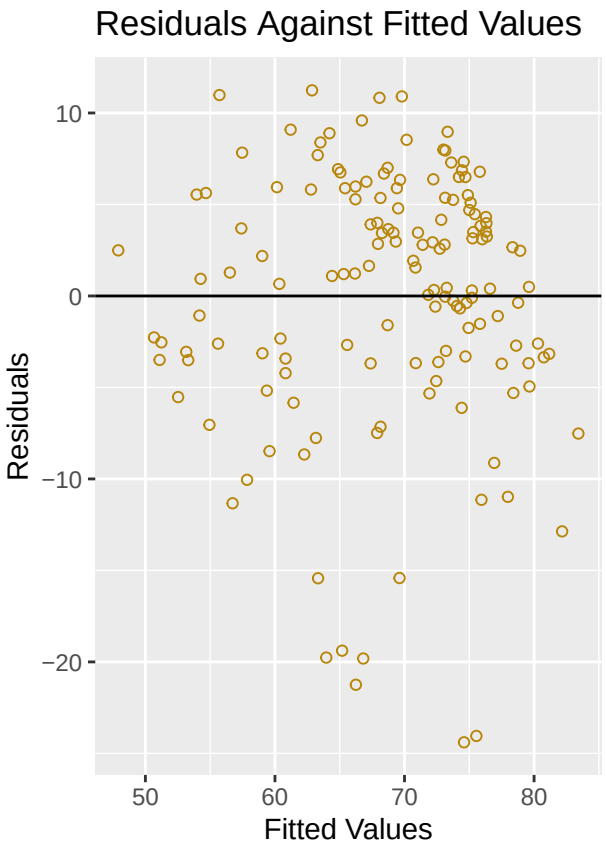
Life Expectancy ~ log(CO2)

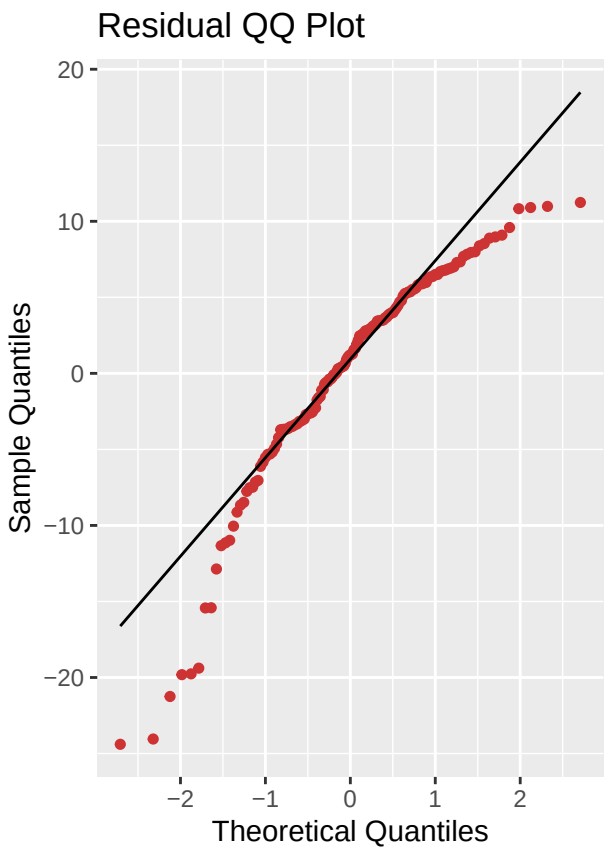
Table 13: Life Expectancy ~ log(CO2)

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	65.423520	0.6397774	102.25981	0
log(CO2)	4.622899	0.3473506	13.30903	0

```
## Saving 6.5 x 4.5 in image
## `geom_smooth()` using formula = 'y ~ x'
## Saving 6.5 x 4.5 in image
## Saving 6.5 x 4.5 in image
## Saving 6.5 x 4.5 in image
## `geom_smooth()` using formula = 'y ~ x'
```


Life Expectancy ~ log(CO2)
Life Expectancy
80
70
60
50
−4
−2
0
2
4
log(CO2)


Residuals Against Fitted Values
Residuals
10
0
−10
−20
50
60
70
80
Fitted Values

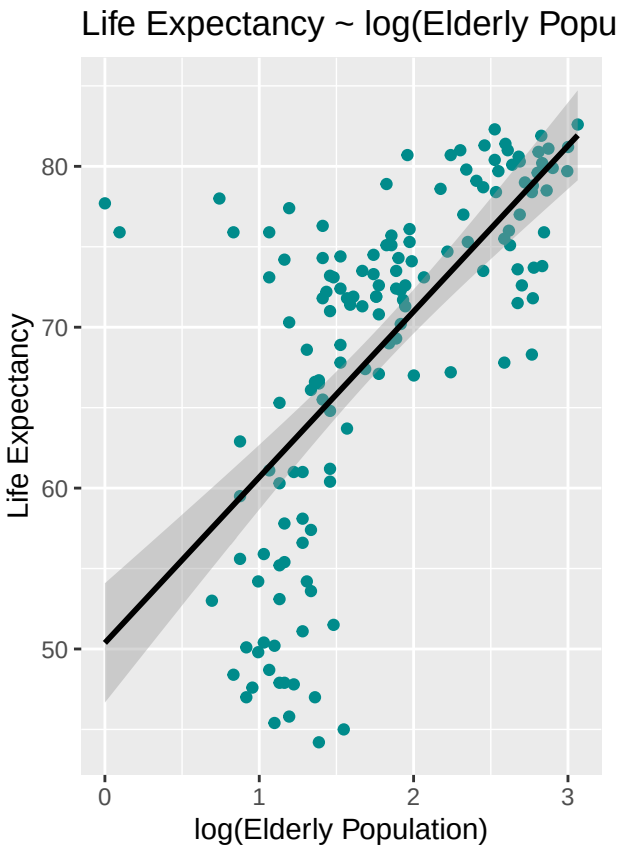


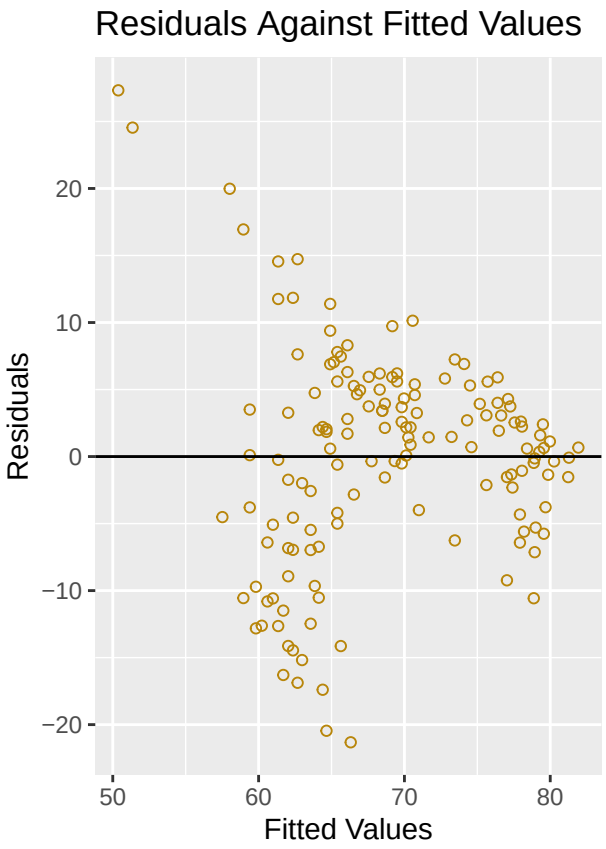
Life Expectancy ~ log(Elderly Population)

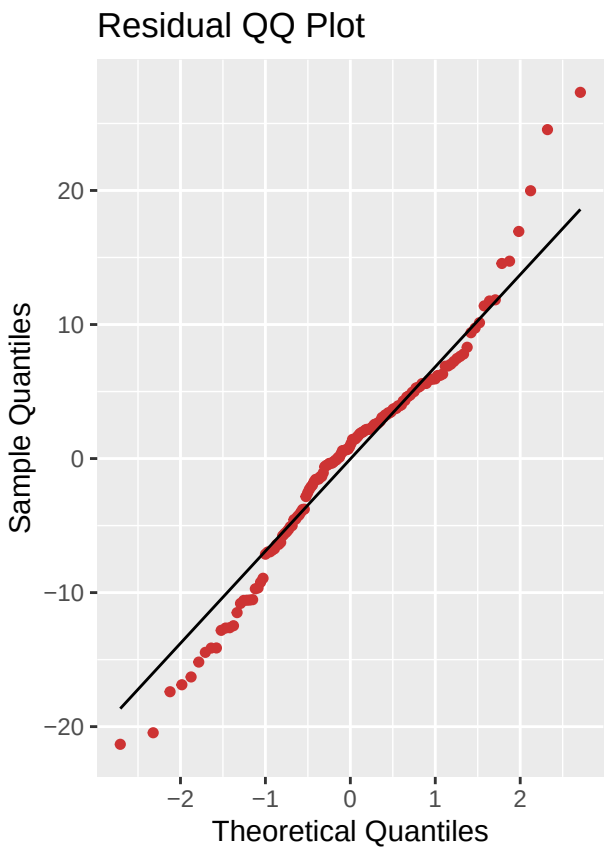
Table 14: Life Expectancy ~ log(Elderly Population)

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	50.37800	1.8762682	26.85011	0
log(Elderly Population)	10.29956	0.9816733	10.49184	0

```
## Saving 6.5 x 4.5 in image
## `geom_smooth()` using formula = 'y ~ x'
## Saving 6.5 x 4.5 in image
## Saving 6.5 x 4.5 in image
## Saving 6.5 x 4.5 in image
## `geom_smooth()` using formula = 'y ~ x'
```


Life Expectancy ~ log(Elderly Popu
Life Expectancy
80
70
60
50
0
1
2
3
log(Elderly Population)


Residuals Against Fitted Values
Residuals
20
10
0
−10
−20
50
60
70
80
Fitted Values

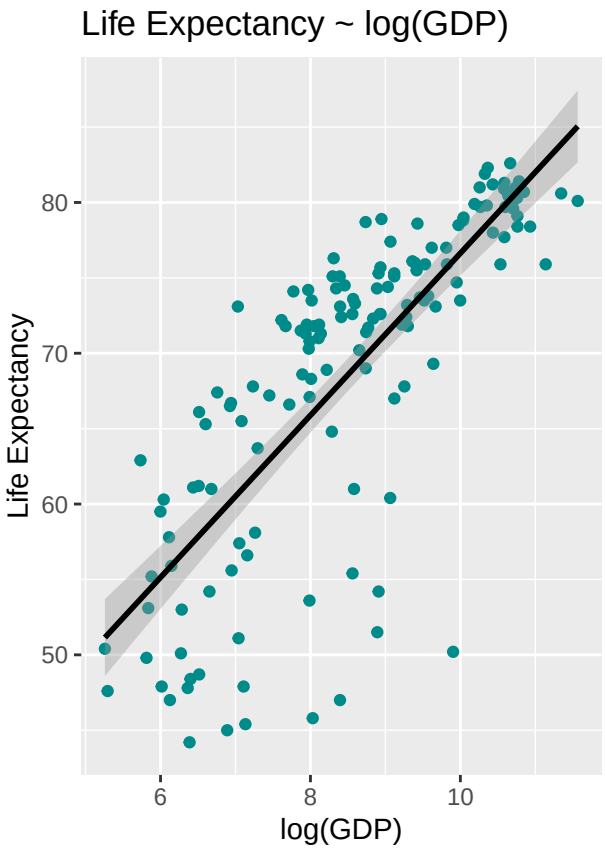


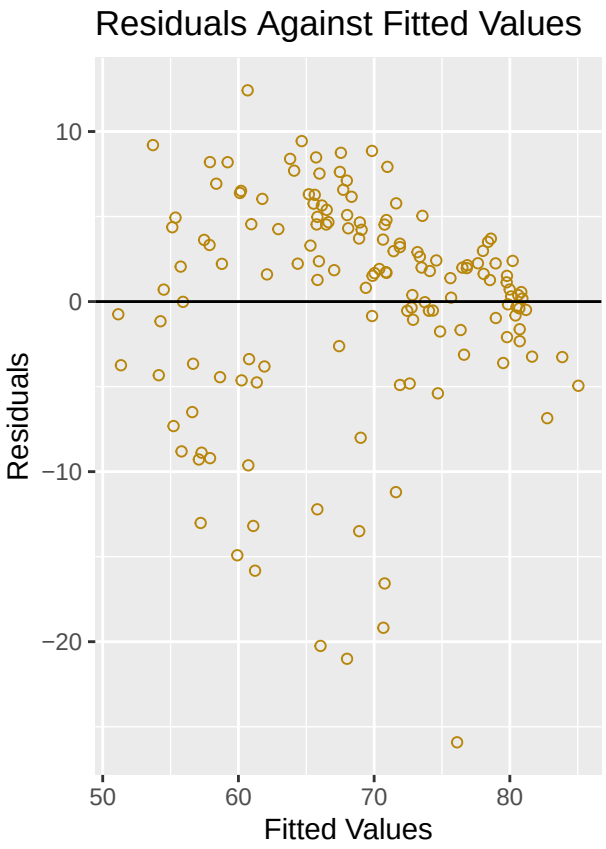
Life Expectancy ~ log(GDP)

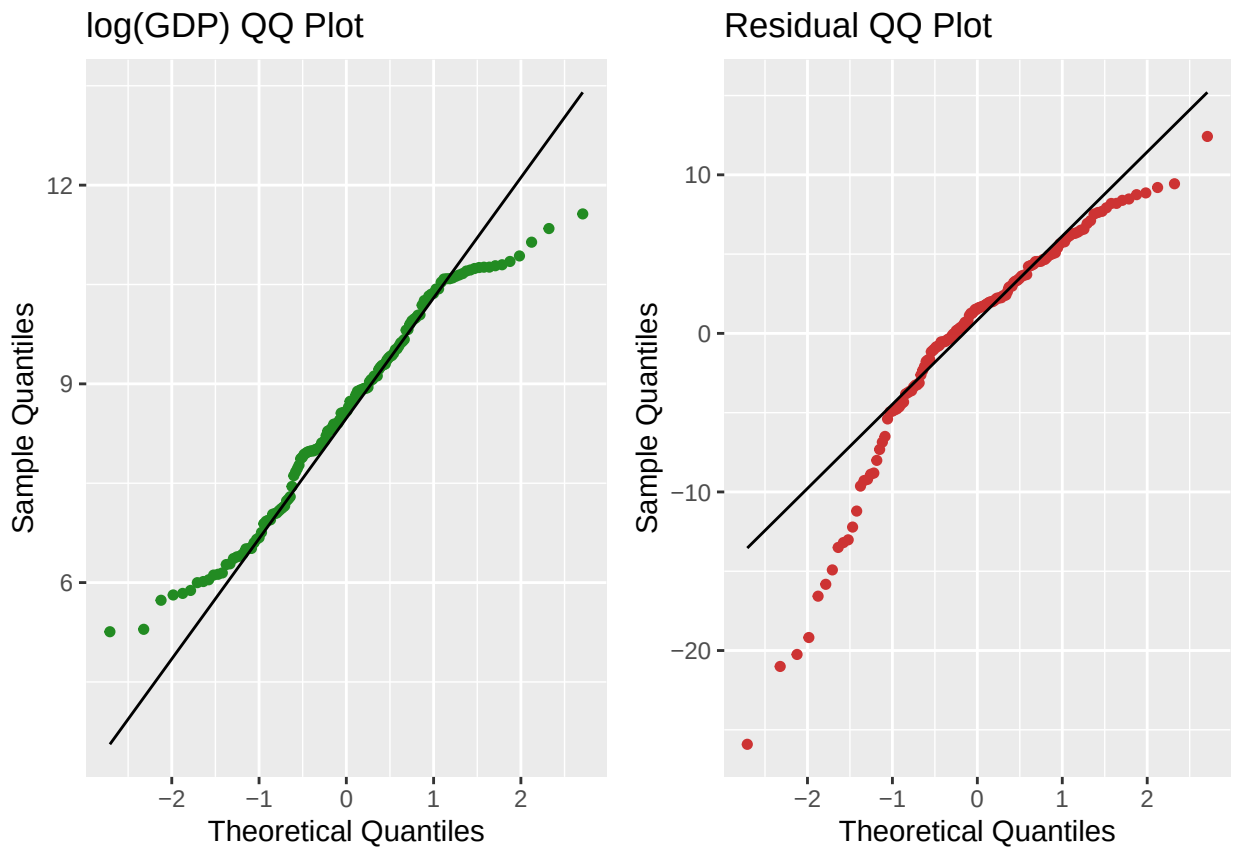
Table 15: Life Expectancy ~ log(GDP)

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	22.885285	3.0930132	7.399026	0
log(GDP)	5.374861	0.3563291	15.083980	0

```
## Saving 6.5 x 4.5 in image
## `geom_smooth()` using formula = 'y ~ x'
## Saving 6.5 x 4.5 in image
## Saving 6.5 x 4.5 in image
## Saving 6.5 x 4.5 in image
## `geom_smooth()` using formula = 'y ~ x'
```


Life Expectancy ~ log(GDP)
Life Expectancy
80
70
60
50
6
8
10
log(GDP)


Residuals Against Fitted Values
Residuals
10
0
−10
−20
50
60
70
80
Fitted Values

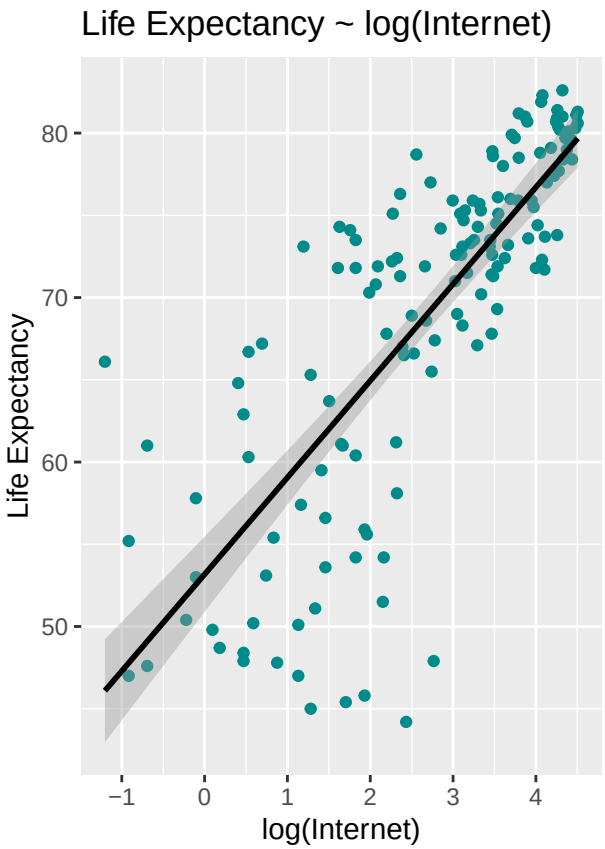


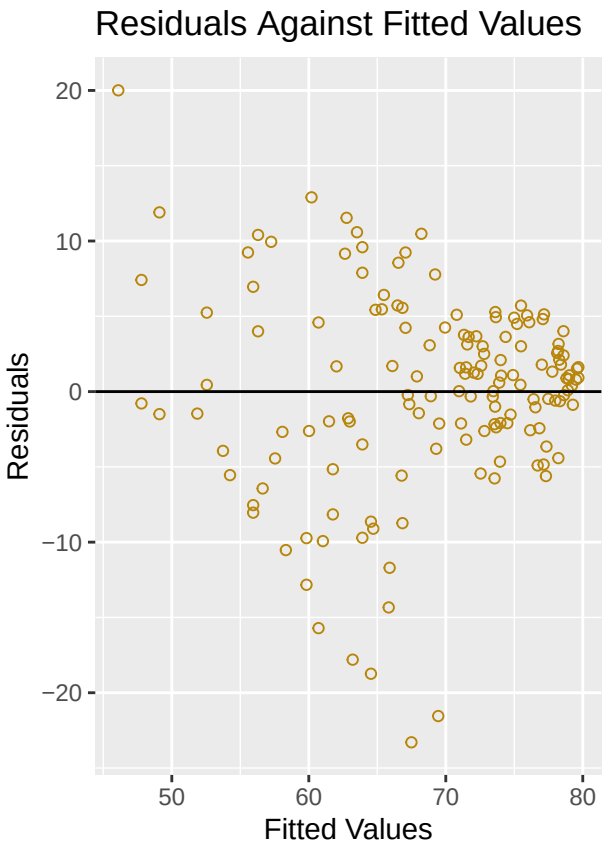
Life Expectancy ~ log(Internet)

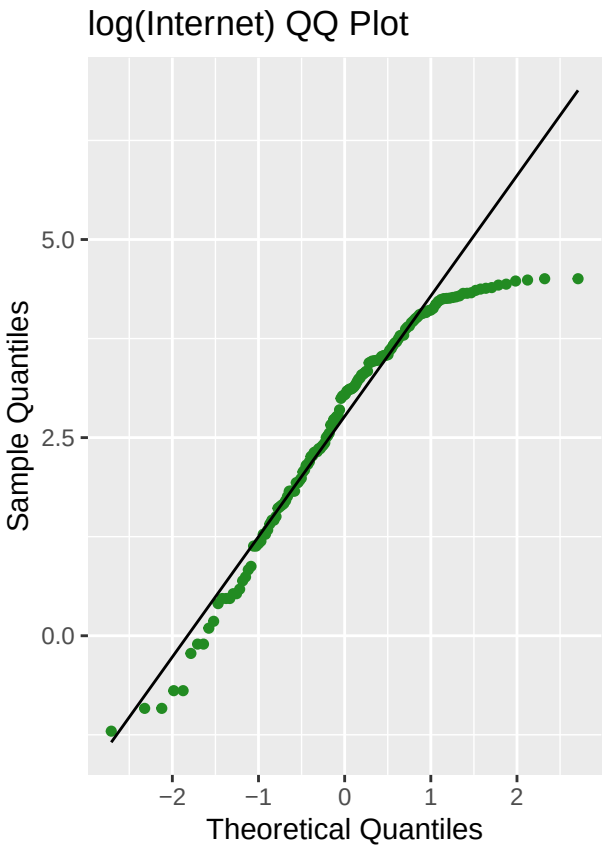
Table 16: Life Expectancy ~ log(Internet)

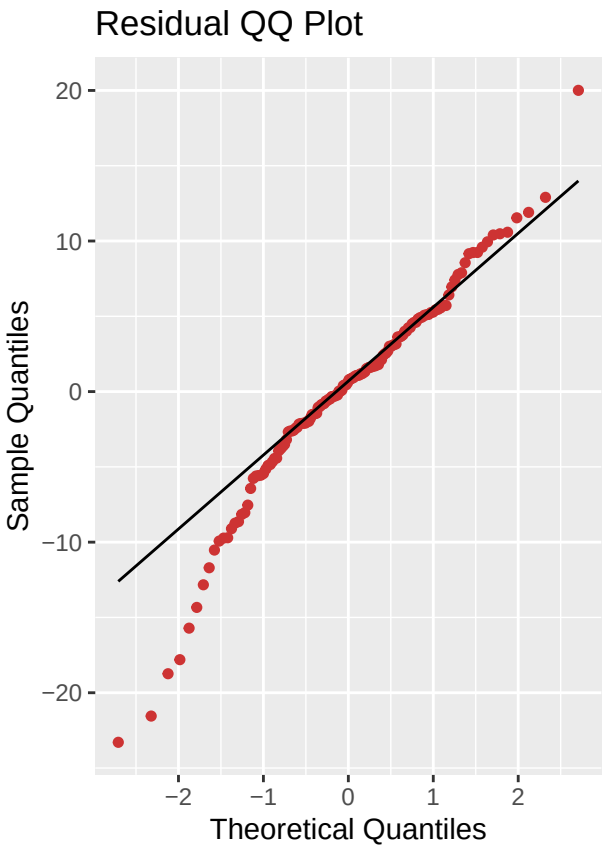
	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	53.175799	1.1599814	45.84194	0
log(Internet)	5.882667	0.3858553	15.24579	0

```
## Saving 6.5 x 4.5 in image
## `geom_smooth()` using formula = 'y ~ x'
## Saving 6.5 x 4.5 in image
## Saving 6.5 x 4.5 in image
## Saving 6.5 x 4.5 in image
## `geom_smooth()` using formula = 'y ~ x'
```


Life Expectancy ~ log(Internet)
Life Expectancy
80
70
60
50
−1
0
1
2
3
4
log(Internet)


Residuals Against Fitted Values
Residuals
20
10
0
−10
−20
50
60
70
80
Fitted Values





State All Rsq

Table 17: R Squared Values

XVar	Rsq	Adj.Rsq	Trans.Rsq
Birth Rate	0.7417423	0.7399734	NA
Cell	0.4306327	0.4267329	NA
CO2	0.1893512	0.1837988	0.5481698
Elderly Population	0.3978775	0.3937534	0.4298627
GDP	0.3367658	0.3322231	0.6091308
Health	0.0873995	0.0811488	NA
Internet	0.5127019	0.5093643	0.6141996
Land Area	0.0016413	-0.0051968	NA
Population	0.0000382	-0.0068109	NA
Rural	0.3855977	0.3813895	NA

Multifactor Models

Table 18: Forward Selection Predictiors

	Include
(Intercept)	TRUE
population	FALSE
rural	TRUE

	Include
health	TRUE
internet	TRUE
birth_rate	TRUE
elderly_pop	TRUE
co2	FALSE
gdp	FALSE
cell	FALSE

Table 19: Forward Selection Algorithm | nbest=5

	population	rural	health	internet	birth_rate	elderly_pop	co2	gdp	cell
1 (1)					*				
2 (1)				*	*				
3 (1)		*		*	*				
4 (1)		*	*	*	*				
5 (1)		*	*	*	*	*			
6 (1)		*	*	*	*	*			*
7 (1)		*	*	*	*	*	*		*
8 (1)		*	*	*	*	*	*	*	*

Table 20: Backward Elimination Predictiors

	Include
(Intercept)	TRUE
population	FALSE
rural	TRUE
health	TRUE
internet	TRUE
birth_rate	TRUE
elderly_pop	TRUE
co2	FALSE
gdp	FALSE
cell	FALSE

Table 21: Backward Elimination Algorithm | nbest=5

	population	rural	health	internet	birth_rate	elderly_pop	co2	gdp	cell
1 (1)					*				
2 (1)				*	*				
3 (1)		*		*	*				
4 (1)		*	*	*	*				
5 (1)		*	*	*	*	*			
6 (1)		*	*	*	*	*		*	
7 (1)		*	*	*	*	*	*	*	
8 (1)		*	*	*	*	*	*	*	*

Assess Multicollinearity

```
## Warning: `select_()` was deprecated in dplyr 0.7.0.
## i Please use `select()` instead.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

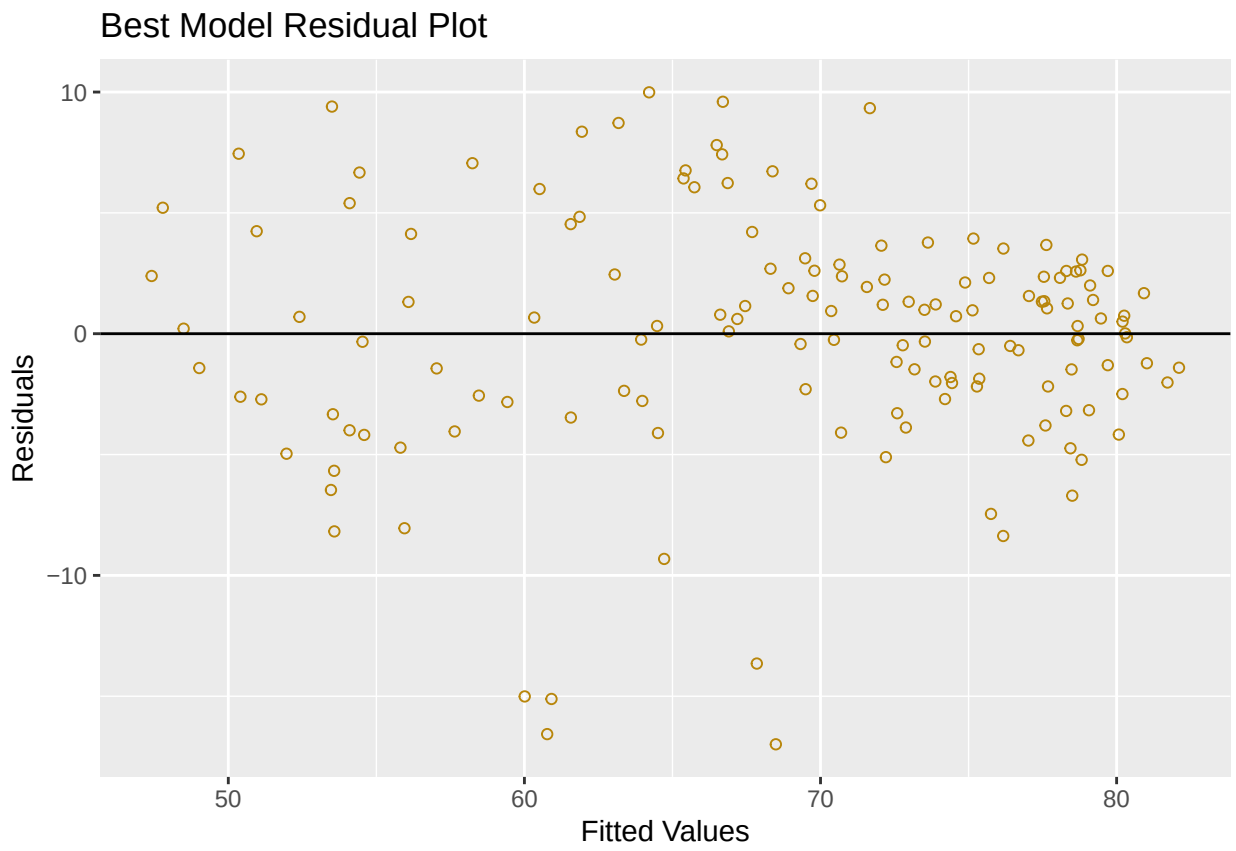
Table 22: VIF Values

	LandArea	Population	Rural	Health	Internet	BirthRate	ElderlyPop	CO2	GDP	Cell
LandArea	Inf	1.260	1.022	1.002	1.009	1.010	1.016	1.023	1.011	1.002
Population	1.260	Inf	1.006	1.010	1.000	1.004	1.001	1.001	1.002	1.009
Rural	1.022	1.006	Inf	1.023	1.775	1.550	1.231	1.791	2.306	1.675
Health	1.002	1.010	1.023	Inf	1.084	1.062	1.161	1.008	1.093	1.009
Internet	1.009	1.000	1.775	1.084	Inf	2.672	1.750	2.659	3.283	1.999
BirthRate	1.010	1.004	1.550	1.062	2.672	Inf	2.549	2.973	2.799	1.823
ElderlyPop	1.016	1.001	1.231	1.161	1.750	2.549	Inf	1.401	1.627	1.270
CO2	1.023	1.001	1.791	1.008	2.659	2.973	1.401	Inf	4.304	2.087
GDP	1.011	1.002	2.306	1.093	3.283	2.799	1.627	4.304	Inf	2.175
Cell	1.002	1.009	1.675	1.009	1.999	1.823	1.270	2.087	2.175	Inf

Best Model

Table 23: Best Model Summary

	Estimate	Std. Error	t value	Pr(>|t|)	RSq
(Intercept)	82.2440751	3.8475946	21.375452	0.0000000	0.7837944
rural	-0.0460924	0.0245643	-1.876394	0.0626535	NA
health	0.2307352	0.1063916	2.168734	0.0317680	NA
internet	1.5883390	0.5275125	3.010998	0.0030829	NA
birth_rate	-0.7054330	0.0816440	-8.640350	0.0000000	NA
elderly_pop	-1.5119081	1.0341320	-1.462007	0.1459489	NA


Best Model Residual Plot
10
0
−10
Residuals
50
60
70
80
Fitted Values