maternal healthcare

Waleed Akrabi

2024-03-26

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```
# load in country indicators
country_indicators <-
  read_csv("country_indicators.csv") %>%
  select(-...1) %>%  # remove first column
  select(iso3, everything()) %>%  # reorder the columns to put iso3 as column 1
  rename(country_code_iso3 = iso3)  # rename first column to country_code_iso3
```

```
## New names:
## Rows: 218 Columns: 1332
## -- Column specification
## -------------------------------------------------------- Delimiter: "," chr
## (8): iso3, hdr_hdicode, hdr_region, wbi_income_group, wbi_lending_cat... dbl
## (1324): ...1, sowc_demographics__population-thousands-2021_total, sowc_d...
## i Use `spec()` to retrieve the full column specification for this data. i
## Specify the column types or set `show_col_types = FALSE` to quiet this message.
## * `` -> `...1`
```

```
# preview data
country_indicators
```

```
## # A tibble: 218 x 1,331
##    country_code_iso3 sowc_demographics__population-thou~1 sowc_demographics__p~2
##    <chr>                                           <dbl>                  <dbl>
##  1 AFG                                             40099.                 20298.
##  2 ALB                                              2855.                   574.
##  3 DZA                                             44178.                 15526.
##  4 AND                                                79.0                  12.8
##  5 AGO                                             34504.                 17833.
##  6 AIA                                                15.8                   3.29
##  7 ATG                                                93.2                  21.3
##  8 ARG                                             45277.                 12669.
##  9 ARM                                              2791.                   669.
## 10 AUS                                             25921.                  5667.
## # i 208 more rows
## # i abbreviated names: 1: `sowc_demographics__population-thousands-2021_total`,
## #   2: `sowc_demographics__population-thousands-2021_under-18`
## # i 1,328 more variables:
```

```
## #   `sowc_demographics__population-thousands-2021_under-5` <dbl>,
## #   `sowc_demographics__annual-population-growth-rate_2000-2020` <dbl>,
## #   `sowc_demographics__annual-population-growth-rate_2020-2030-a` <dbl>, ...
```

##Hear we clean and get data from country_indicators.csv to get the 3 main indicators we need for our model and pair them to their country codes

```
data_maternal <- country_indicators %>%
   rename(codes = country_code_iso3) %>%
  mutate(
    skilled_birth_attendant =
      (`sowc_maternal-and-newborn-health__delivery-care-2016-2021-r_skilled-birth-attendant`),
    Service_coverage = (`sowc_maternal-and-newborn-health__universal-health-coverage-2019_service-covera
    maternal_mortality = (`sowc_maternal-and-newborn-health__maternal-mortality-2020-c_maternal-mortali
    ) %>%
select(codes,skilled_birth_attendant, Service_coverage, maternal_mortality)

data_maternal
```

```
## # A tibble: 218 x 4
##    codes skilled_birth_attendant Service_coverage maternal_mortality
##    <chr>                   <dbl>            <dbl>              <dbl>
##  1 AFG                      61.8               37             620.
##  2 ALB                      99.8               62               8.28
##  3 DZA                      98.8               75              77.7
##  4 AND                     100                 NA              NA
##  5 AGO                      49.6               39             222.
##  6 AIA                     100                 NA              NA
##  7 ATG                      99                 72              21.2
##  8 ARG                      98.8               73              44.9
##  9 ARM                      99.8               69              27.2
## 10 AUS                      98.8               87               2.94
## # i 208 more rows
```

##Below we add the SDG_index_score for each country as that is our dependent variable we need for our regression model (Y)

```
SDG <- read_csv('sdr_fd5e4b5a.csv')%>%
  rename(sdg_score = `2023 SDG Index Score`, codes = `Country Code ISO3`)%>%
select(codes, sdg_score)
```

```
## New names:
## Rows: 206 Columns: 59
## -- Column specification
## -------------------------------------------------------- Delimiter: "," chr
## (36): Goal 1 Dash, Goal 1 Trend, Goal 2 Dash, Goal 2 Trend, Goal 3 Dash,... dbl
## (23): ...1, Goal 1 Score, Goal 2 Score, Goal 3 Score, Goal 4 Score, Goal...
## i Use `spec()` to retrieve the full column specification for this data. i
## Specify the column types or set `show_col_types = FALSE` to quiet this message.
## * `` -> `...1`
```

```
sdg_indicators_score <- left_join(x = data_maternal, y = SDG, by = "codes")

sdg_indicators_score
```

```
## # A tibble: 218 x 5
##    codes skilled_birth_attendant Service_coverage maternal_mortality sdg_score
```

```
##    <chr>                  <dbl>           <dbl>             <dbl>     <dbl>
##  1 AFG                     61.8              37             620.       49
##  2 ALB                     99.8              62               8.28     73.5
##  3 DZA                     98.8              75              77.7      70.8
##  4 AND                    100                NA              NA        NA
##  5 AGO                     49.6              39             222.       50.8
##  6 AIA                    100                NA              NA        NA
##  7 ATG                     99                72              21.2      NA
##  8 ARG                     98.8              73              44.9      73.7
##  9 ARM                     99.8              69              27.2      73.3
## 10 AUS                     98.8              87               2.94     75.9
## # i 208 more rows
```

```
# load in country indicators
country_codes <-
  read_csv("country_codes.csv")
```

```
## New names:
## Rows: 298 Columns: 125
## -- Column specification
## -------------------------------------------------------- Delimiter: "," chr
## (99): Global Name_en (M49), Region Name_en (M49), Sub-region Name_en (M4... dbl
## (22): ...1, Global Code (M49), Region Code (M49), Intermediate Region Co... lgl
## (4): Sub-region Code (M49), Least Developed Countries (LDC) (M49), Land...
## i Use `spec()` to retrieve the full column specification for this data. i
## Specify the column types or set `show_col_types = FALSE` to quiet this message.
## * `` -> `...1`
```

```
country_indicators
```

```
## # A tibble: 218 x 1,331
##    country_code_iso3 sowc_demographics__population-thou~1 sowc_demographics__p~2
##    <chr>                                           <dbl>                  <dbl>
##  1 AFG                                            40099.                20298.
##  2 ALB                                             2855.                  574.
##  3 DZA                                            44178.                15526.
##  4 AND                                               79.0                  12.8
##  5 AGO                                            34504.                17833.
##  6 AIA                                               15.8                   3.29
##  7 ATG                                               93.2                  21.3
##  8 ARG                                            45277.                12669.
##  9 ARM                                             2791.                  669.
## 10 AUS                                            25921.                 5667.
## # i 208 more rows
## # i abbreviated names: 1: `sowc_demographics__population-thousands-2021_total`,
## #   2: `sowc_demographics__population-thousands-2021_under-18`
## # i 1,328 more variables:
## #   `sowc_demographics__population-thousands-2021_under-5` <dbl>,
## #   `sowc_demographics__annual-population-growth-rate_2000-2020` <dbl>,
## #   `sowc_demographics__annual-population-growth-rate_2020-2030-a` <dbl>, ...
```

##we then get the data regions

```
data_regions <- country_codes %>%
  rename (region = `Region Name_en (M49)`,
          sub_region = `Intermediate Region Name_en (M49)`,
```

```
          codes = `ISO-alpha3 Code (M49)`,
          country = `Country or Area_en (M49)`) %>%
  select(region, sub_region, codes, country)

data_final_unwrangled <- inner_join(x = data_regions, y = sdg_indicators_score, by = "codes")
```

##here we have the final wrangled data that we will use for our method

```
data_final_wrangled <- data_final_unwrangled %>%
  filter(region == "Africa") %>%
  filter(!is.na(skilled_birth_attendant)) %>%
  filter(!is.na(Service_coverage)) %>%
  filter(!is.na(maternal_mortality)) %>%
  filter(!is.na(sdg_score)) %>%
  distinct(.keep_all = TRUE) %>%
  select(region, sub_region, country, skilled_birth_attendant, Service_coverage, maternal_mortality, sdg

glimpse(data_final_wrangled)
```

```
## Rows: 49
## Columns: 7
## $ region                  <chr> "Africa", "Africa", "Africa", "Africa", "Afric~
## $ sub_region              <chr> NA, NA, NA, NA, NA, "Eastern Africa", "Eastern~
## $ country                 <chr> "Algeria", "Egypt", "Morocco", "Sudan", "Tunis~
## $ skilled_birth_attendant <dbl> 98.8, 91.5, 86.6, 77.7, 99.5, 85.1, 82.2, 87.4~
## $ Service_coverage        <dbl> 75, 70, 73, 44, 70, 44, 44, 48, 38, 56, 35, 48~
## $ maternal_mortality      <dbl> 77.69479, 16.81971, 70.83838, 270.35535, 36.62~
## $ sdg_score               <dbl> 70.8, 69.6, 70.9, 48.6, 72.5, 53.9, 51.7, 52.7~
```

##Below we now implement a muliple linear regression model on our dependent and independent variables and get a summary

```
# Define dependent variable (Y)
Y <- data_final_wrangled$sdg_score

# Define independent variables (X)
X <- data_final_wrangled %>%
  select(skilled_birth_attendant, Service_coverage, maternal_mortality)

# Fit multiple linear regression model
model <- lm(Y ~ ., data = X)

summary(model)
```

```
##
## Call:
## lm(formula = Y ~ ., data = X)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.7706 -2.8382  0.2553  2.3491  7.5460
##
## Coefficients:
##                           Estimate Std. Error t value Pr(>|t|)
## (Intercept)             35.112223   4.205483   8.349 1.08e-10 ***
## skilled_birth_attendant  0.015516   0.039166   0.396    0.694
```

```
## Service_coverage         0.460668   0.071797   6.416 7.51e-08 ***
## maternal_mortality      -0.004647   0.003013  -1.543    0.130
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.656 on 45 degrees of freedom
## Multiple R-squared:  0.7761, Adjusted R-squared:  0.7612
## F-statistic:    52 on 3 and 45 DF,  p-value: 1.146e-14
```

```
summary(model)$coefficients
```

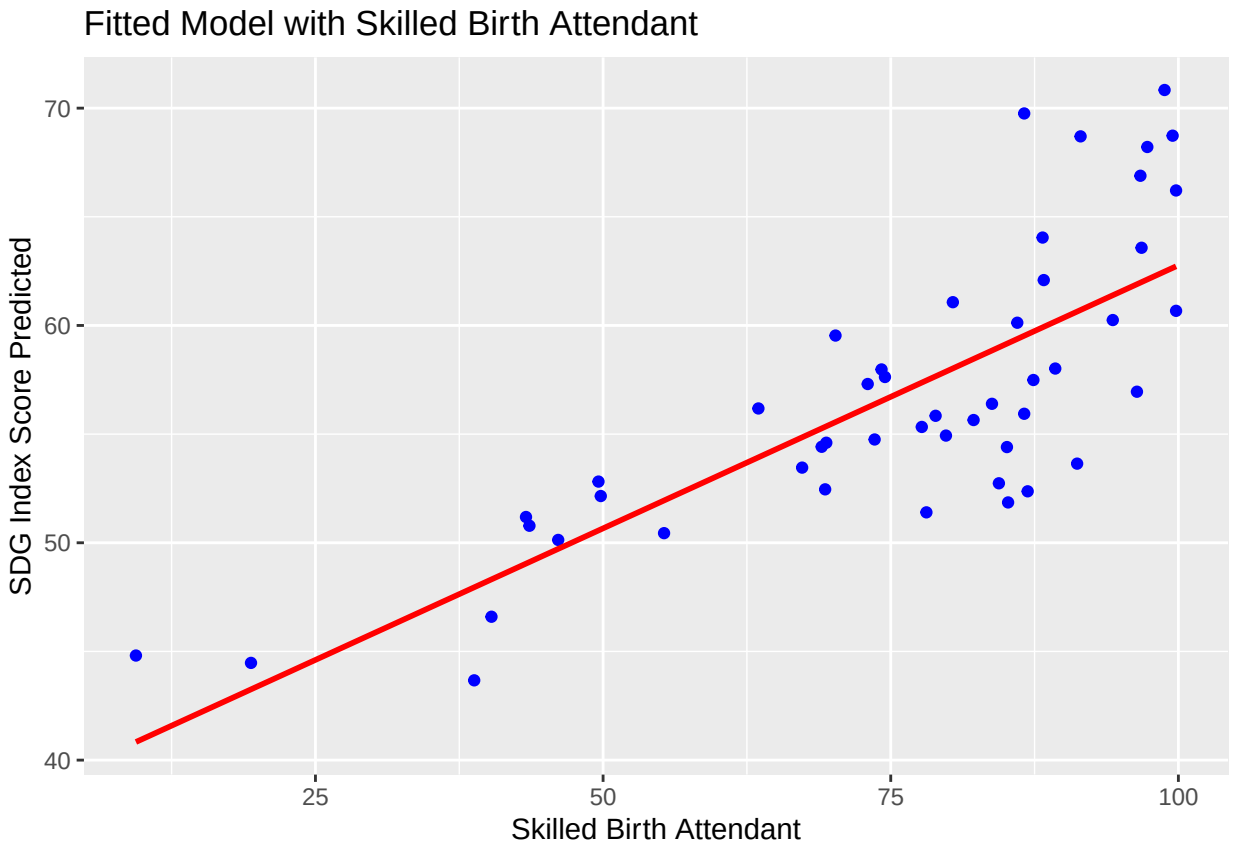
```
##                              Estimate  Std. Error    t value     Pr(>|t|)
## (Intercept)             35.112222716 4.205483162  8.3491531 1.075691e-10
## skilled_birth_attendant  0.015516013 0.039166312  0.3961571 6.938614e-01
## Service_coverage         0.460668314 0.071797218  6.4162419 7.511021e-08
## maternal_mortality      -0.004647399 0.003012677 -1.5426142 1.299282e-01
```

Vizualiation

##First Data visualization is scatter plot with a fitted regression line for each independent variable:

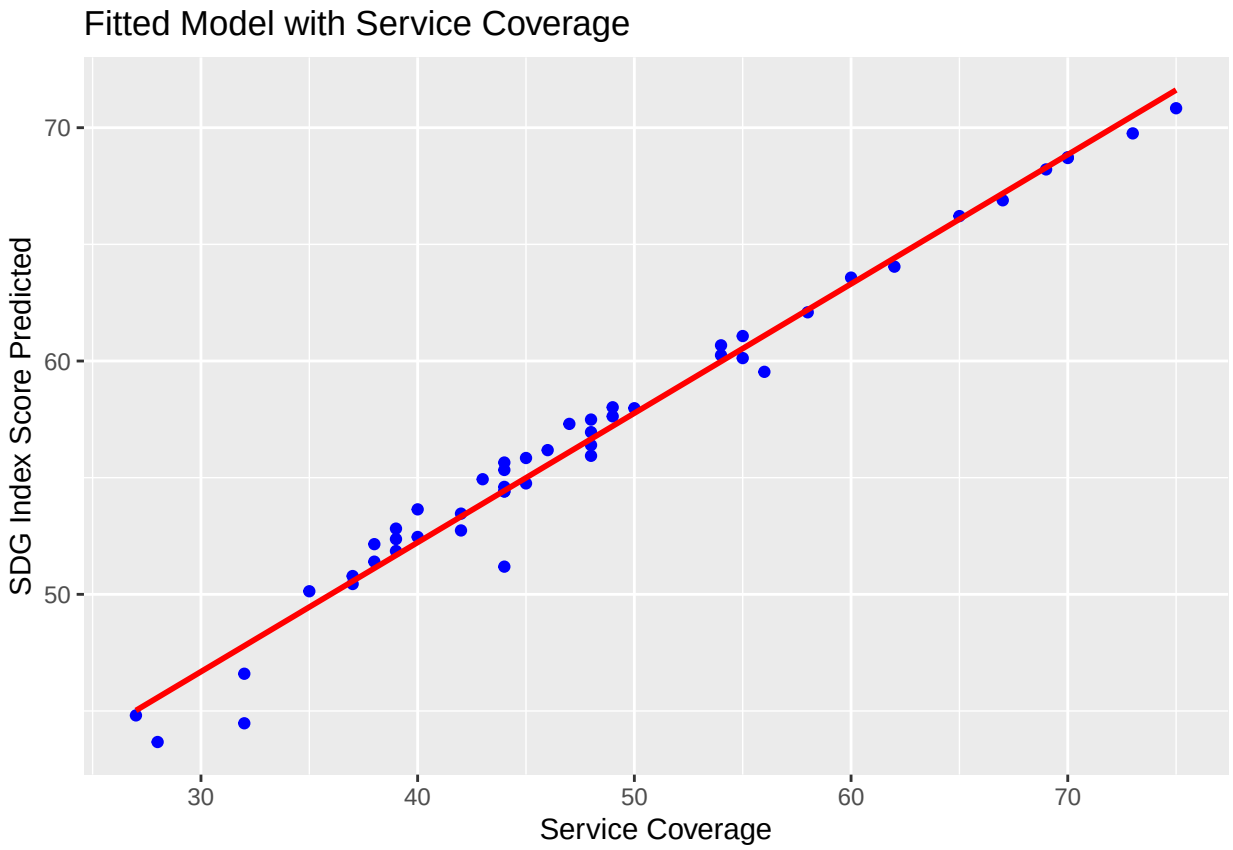
```
# Create a dataframe with the independent variables and their predicted values
plot_data <- data.frame(
  skilled_birth_attendant = data_final_wrangled$skilled_birth_attendant,
  Service_coverage = data_final_wrangled$Service_coverage,
  maternal_mortality = data_final_wrangled$maternal_mortality,
  sdg_score_predicted = predict(model)
)

# Plot for skilled_birth_attendant vs. sdg_score with fitted line
plot_skilled_birth_attendant <- ggplot(plot_data, aes(x = skilled_birth_attendant, y = sdg_score_predic
  geom_point(aes(y = sdg_score_predicted), color = "blue") +
  geom_smooth(method = "lm", se = FALSE, color = "red") +
  labs(x = "Skilled Birth Attendant", y = "SDG Index Score Predicted", title = "Fitted Model with Skill

# Plot for Service_coverage vs. sdg_score with fitted line
plot_Service_coverage <- ggplot(plot_data, aes(x = Service_coverage, y = sdg_score_predicted)) +
  geom_point(aes(y = sdg_score_predicted), color = "blue") +
  geom_smooth(method = "lm", se = FALSE, color = "red") +
  labs(x = "Service Coverage", y = "SDG Index Score Predicted", title = "Fitted Model with Service Cove

# Plot for maternal_mortality vs. sdg_score with fitted line
plot_maternal_mortality <- ggplot(plot_data, aes(x = maternal_mortality, y = sdg_score_predicted)) +
  geom_point(aes(y = sdg_score_predicted), color = "blue") +
  geom_smooth(method = "lm", se = FALSE, color = "red") +
  labs(x = "Maternal Mortality", y = "SDG Index Score Predicted", title = "Fitted Model with Maternal M
plot_skilled_birth_attendant
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



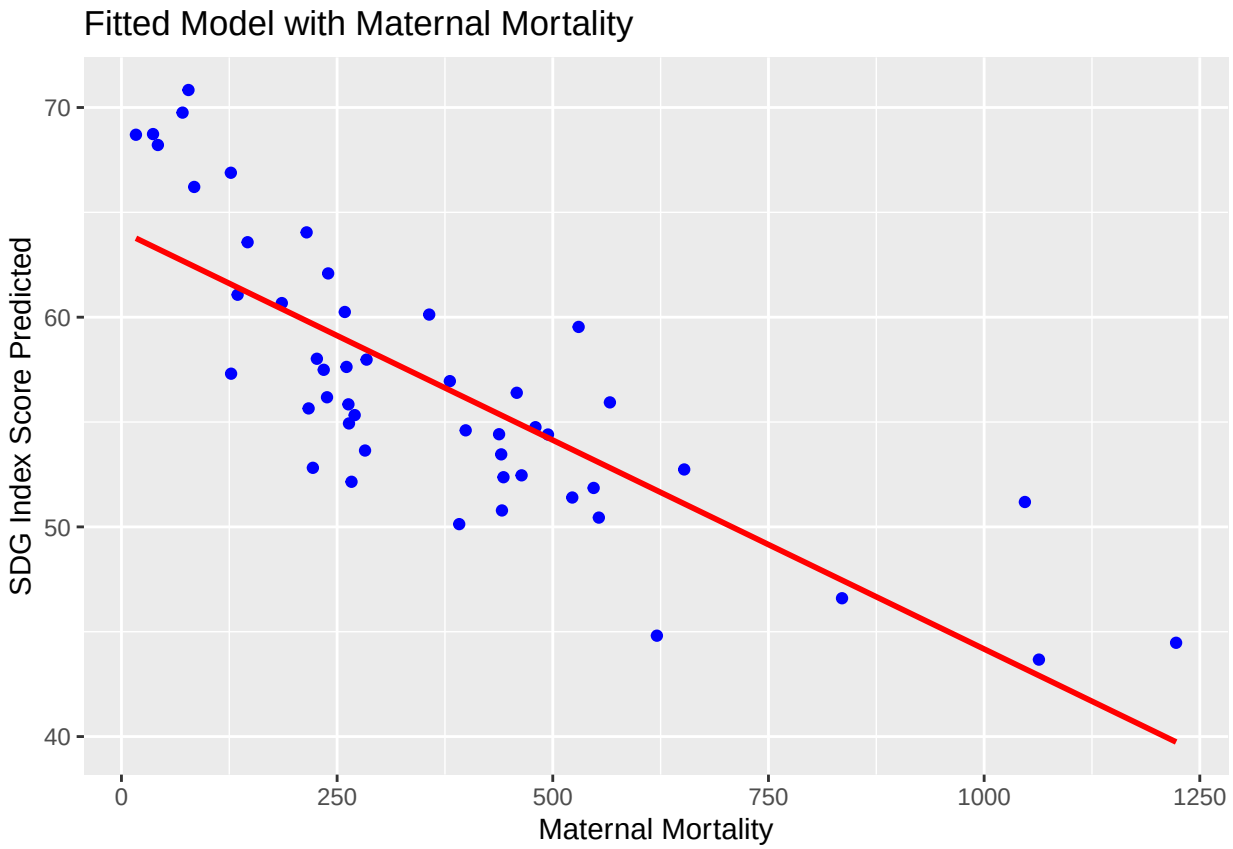

```
plot_Service_coverage
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



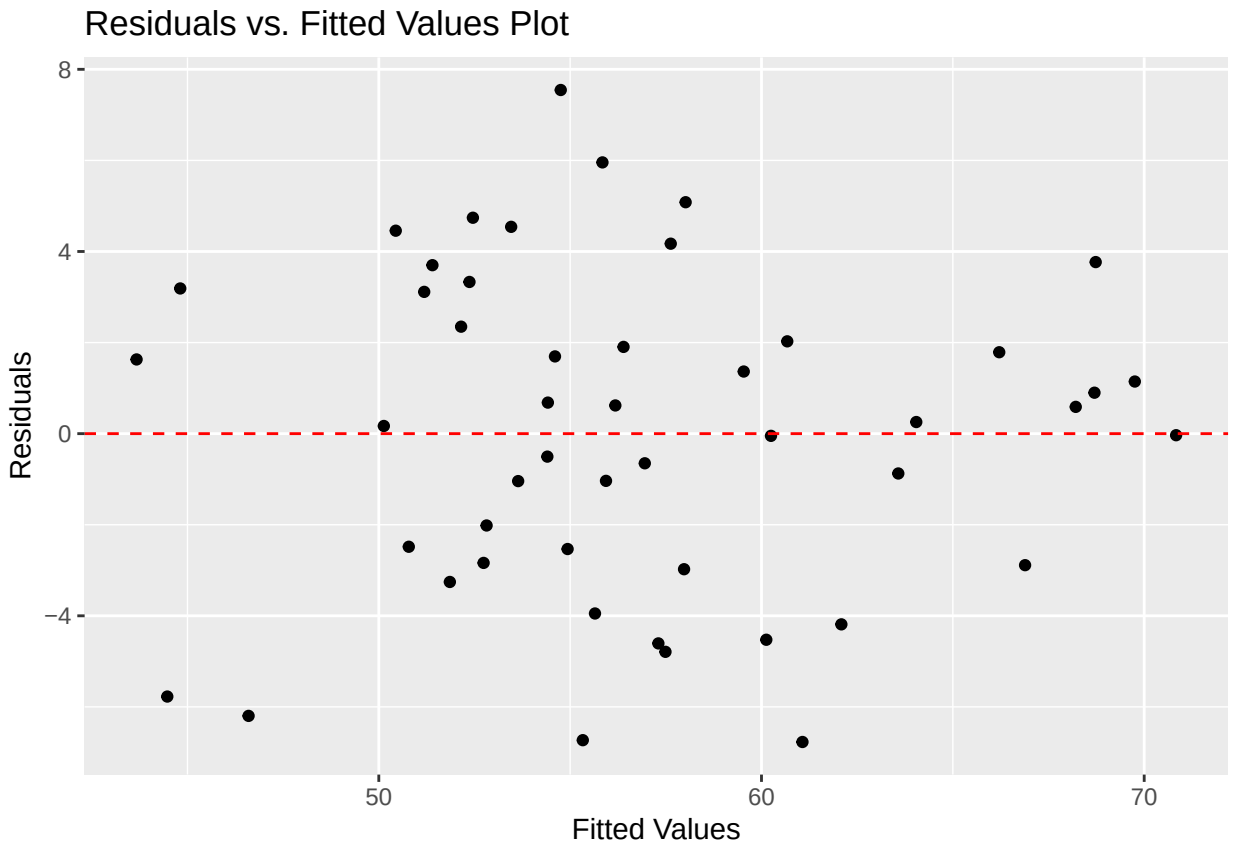

```
plot_maternal_mortality
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



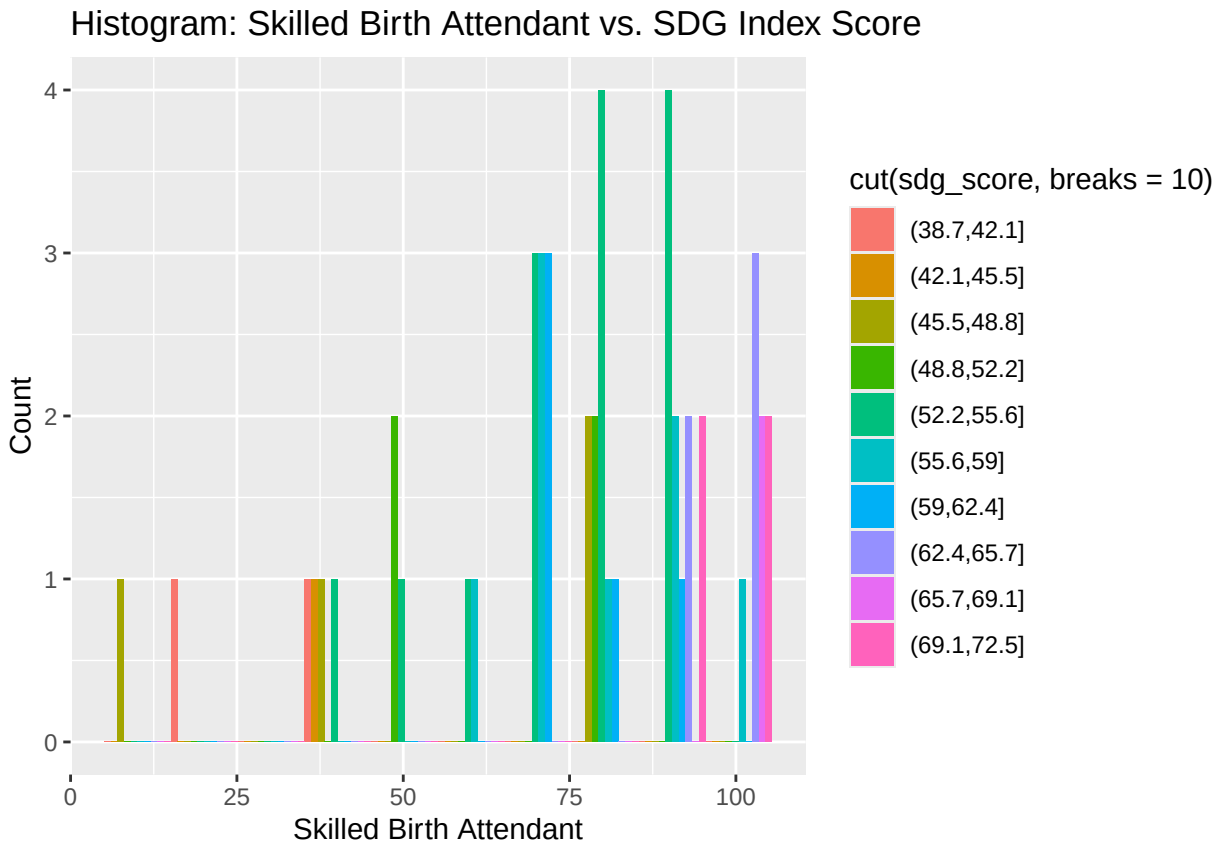
##Another visualization we used is a plot of the residuals versus the fitted values. This plot helps to assess the assumptions of homoscedasticity (constant variance of residuals) and linearity of the model

```r
# Obtain the residuals from the model
residuals <- residuals(model)

# Obtain the fitted values from the model
fitted_values <- fitted(model)

# Create a dataframe with residuals and fitted values
residuals_df <- data.frame(residuals = residuals, fitted_values = fitted_values)

# Plot residuals versus fitted values
residuals_plot <- ggplot(residuals_df, aes(x = fitted_values, y = residuals)) +
  geom_point() +
  geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
  labs(x = "Fitted Values", y = "Residuals", title = "Residuals vs. Fitted Values Plot")

# Display the plot
print(residuals_plot)
```

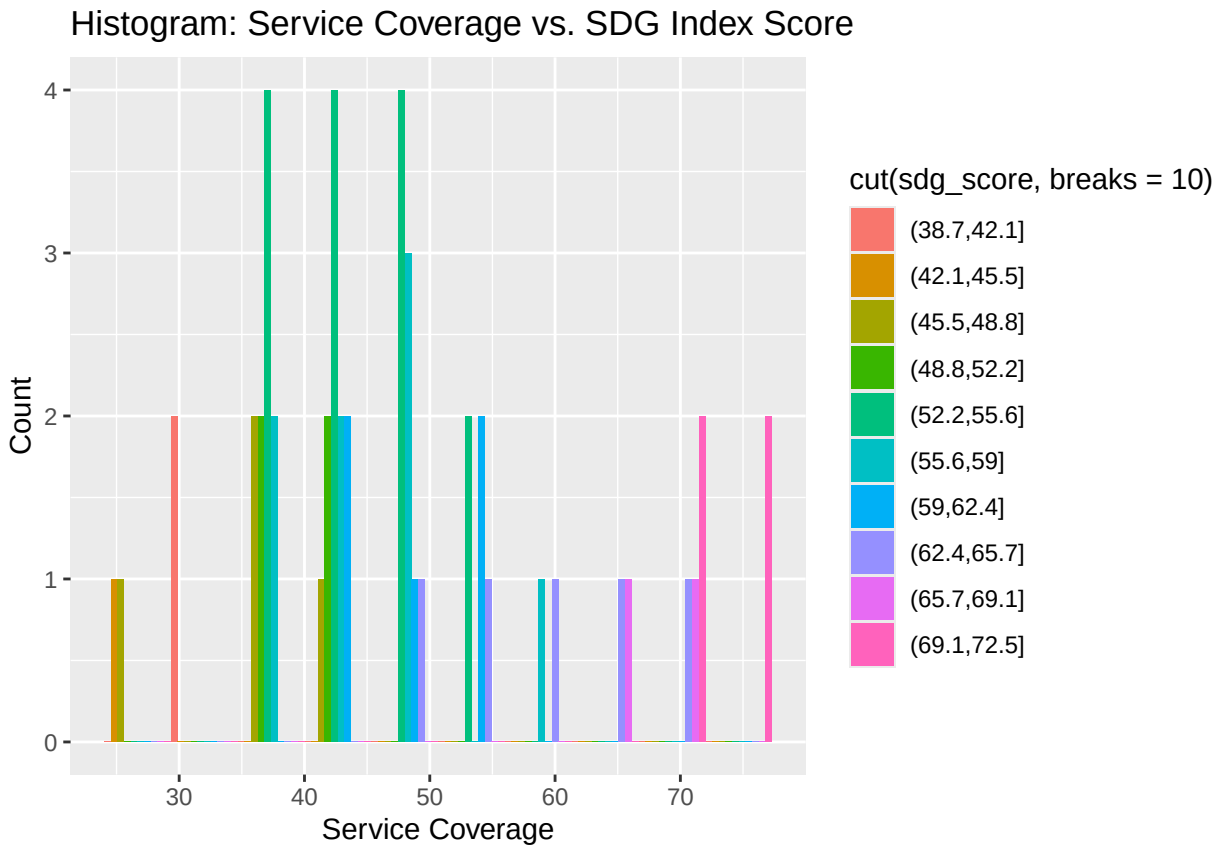


##We can use a histogram to further see the amount of countries that are in each indacator and there corisponding Y value color coded

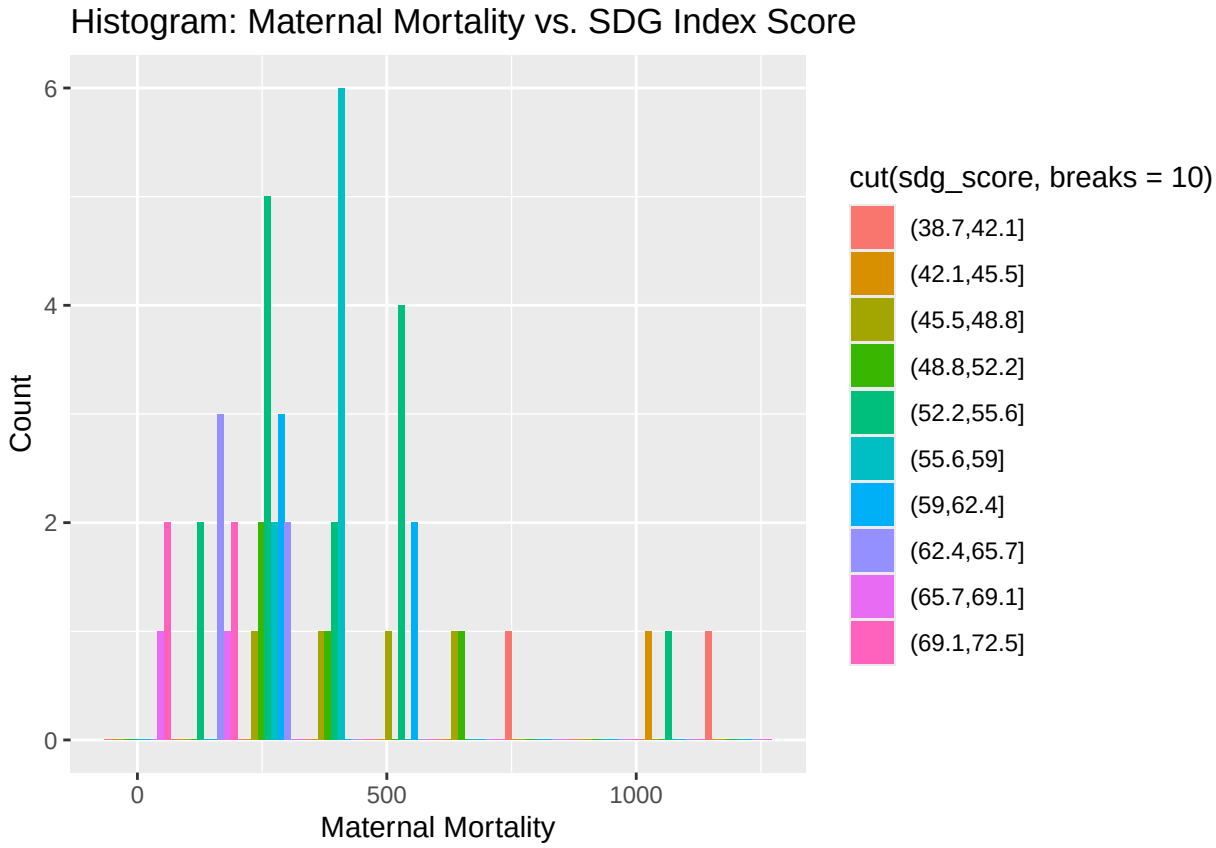
```
# Create histogram for skilled_birth_attendant vs. sdg_score
plot_skilled_birth_attendant <- ggplot(data_final_wrangled, aes(x = skilled_birth_attendant, fill = cut
  geom_histogram(position = "dodge", bins = 10) +
  labs(x = "Skilled Birth Attendant", y = "Count", title = "Histogram: Skilled Birth Attendant vs. SDG

# Create histogram for Service_coverage vs. sdg_score
plot_Service_coverage <- ggplot(data_final_wrangled, aes(x = Service_coverage, fill = cut(sdg_score, br
  geom_histogram(position = "dodge", bins = 10) +
  labs(x = "Service Coverage", y = "Count", title = "Histogram: Service Coverage vs. SDG Index Score")

# Create histogram for maternal_mortality vs. sdg_score
plot_maternal_mortality <- ggplot(data_final_wrangled, aes(x = maternal_mortality, fill = cut(sdg_score
  geom_histogram(position = "dodge", bins = 10) +
  labs(x = "Maternal Mortality", y = "Count", title = "Histogram: Maternal Mortality vs. SDG Index Scor

plot_skilled_birth_attendant
```




```
plot_Service_coverage
```




```
plot_maternal_mortality
```


Histogram: Maternal Mortality vs. SDG Index Score
Count
Maternal Mortality
0
2
4
6
0
500
1000
cut(sdg_score, breaks = 10)
(38.7,42.1]
(42.1,45.5]
(45.5,48.8]
(48.8,52.2]
(52.2,55.6]
(55.6,59]
(59,62.4]
(62.4,65.7]
(65.7,69.1]
(69.1,72.5]