

SDGs of Asia UNICEF Project

2024-04-07

Contents

```
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`
```

```
# 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`
```

```
# load SDG data
```

```
sdg <-
  read_csv("sdr_fd5e4b5a.csv") %>%
  select(-...1) # remove first column
```

```
## 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`
```

```

# rename columns
names(sdg)[1:(2*17)] <-
  paste(c(rep(paste("goal_", 1:17, sep=""), each=2)),
        rep(c("_status", "_trend"), times=17), sep="")
names(sdg)[(2*17 + 1):(3*17)] <-
  paste("goal_", 1:17, "_score", sep="")
names(sdg)[names(sdg)=="2023 SDG Index Score"] <-
  "SDG_index_score_2023"
names(sdg)[names(sdg)=="2023 SDG Index Rank"] <-
  "SDG_index_rank_2023"
names(sdg)[names(sdg)=="Percentage missing values"] <-
  "percentage_missing_values"
names(sdg)[names(sdg)=="International Spillovers Score (0-100)"] <-
  "international_spillover_score"
names(sdg)[names(sdg)=="International Spillovers Rank"] <-
  "international_spillover_rank"
names(sdg)[names(sdg)=="Country Code ISO3"] <-
  "country_code_iso3"

#DATA FORMING
# join tables
data <- inner_join(x=country_indicators, y=sdg, by="country_code_iso3")
country_codes_modified <- rename(country_codes, country_code_iso3 = `ISO-alpha3 Code (M49)`)
final_data <- inner_join(x=data, y=country_codes_modified, by="country_code_iso3")
#filter out Asia contries
data_Asia <- final_data %>% filter(`Region Code (M49)`==142)
#k-means algorithm can not deal with NA/NaN/Inf in SDG_index_score_2023, so we need to clean the data f
clean_data_Asia <- data_Asia %>%
  distinct(country_code_iso3, .keep_all = TRUE) %>% #drop dup
  filter(!is.na(SDG_index_score_2023) & SDG_index_score_2023 != Inf & SDG_index_score_2023 != -Inf)
final_data_clean <- final_data %>%
  distinct(country_code_iso3, .keep_all = TRUE) %>% #drop dup
  filter(!is.na(SDG_index_score_2023) & SDG_index_score_2023 != Inf & SDG_index_score_2023 != -Inf)

##Research Question 1 Part
#Clustering with k-means algorithm
#set seed
set.seed(1010315026)
#k-means algorithm
k <- 4
clustering <- kmeans(clean_data_Asia$SDG_index_score_2023, k)
clustering

## K-means clustering with 4 clusters of sizes 2, 18, 14, 11
##
## Cluster means:
##      [,1]
## 1 47.90000
## 2 69.73889
## 3 63.62143
## 4 74.59091
##
## Clustering vector:
##  [1] 1 4 4 3 3 4 3 3 2 4 4 3 2 2 3 4 4 2 2 3 4 3 2 2 2 3 3 3 2 3 2 3 4 2 2 2 3 2

```

```
## [39] 4 2 2 2 4 1 2
##
## Within cluster sum of squares by cluster:
## [1] 2.42000 38.26278 90.56357 50.14909
## (between_SS / total_SS = 89.9 %)
##
## Available components:
##
## [1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss"
## [6] "betweenss"    "size"         "iter"         "ifault"

```

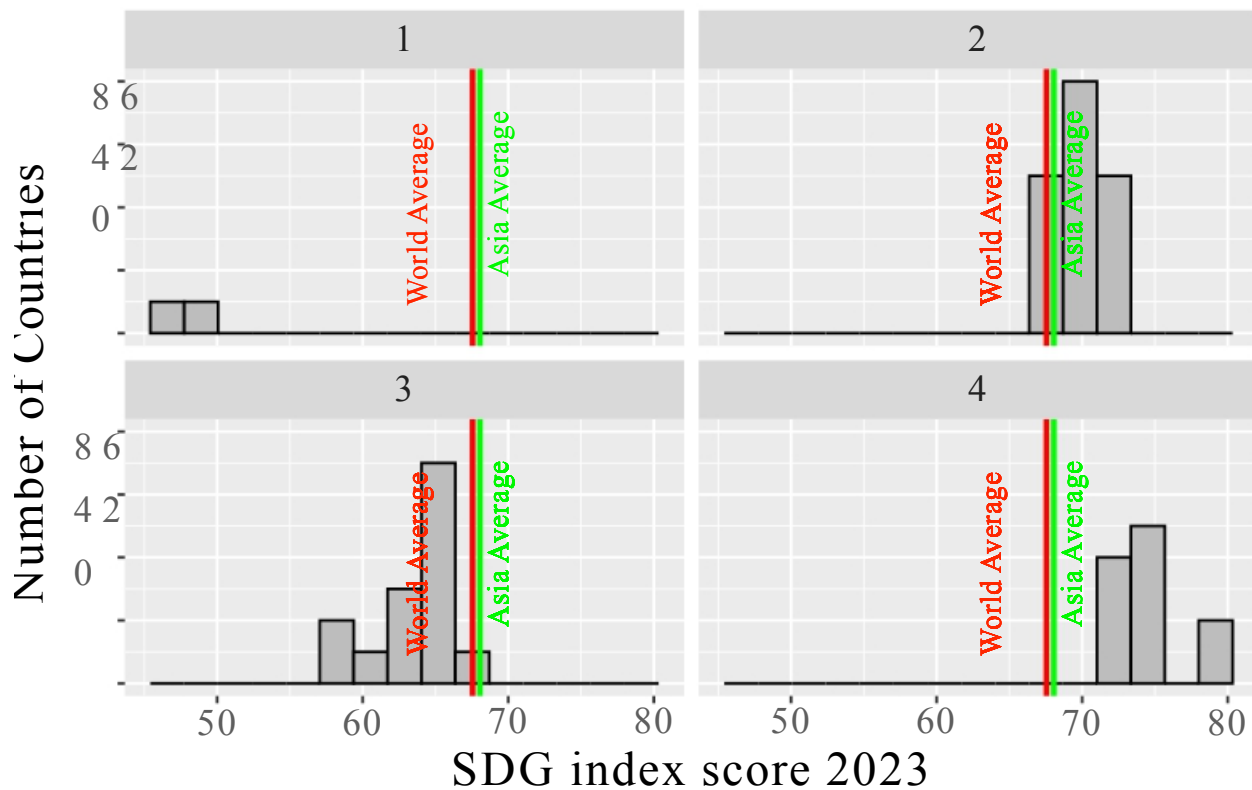
```
# add clustering values to our original dataset
clean_data_Asia <-
  clean_data_Asia %>%
    mutate(cluster = clustering$cluster)

#avg_sdg_index_score
world_avg_sdg_index_score <- mean(final_data_clean$SDG_index_score_2023)
asia_avg_sdg_index_score <- mean(clean_data_Asia$SDG_index_score_2023)

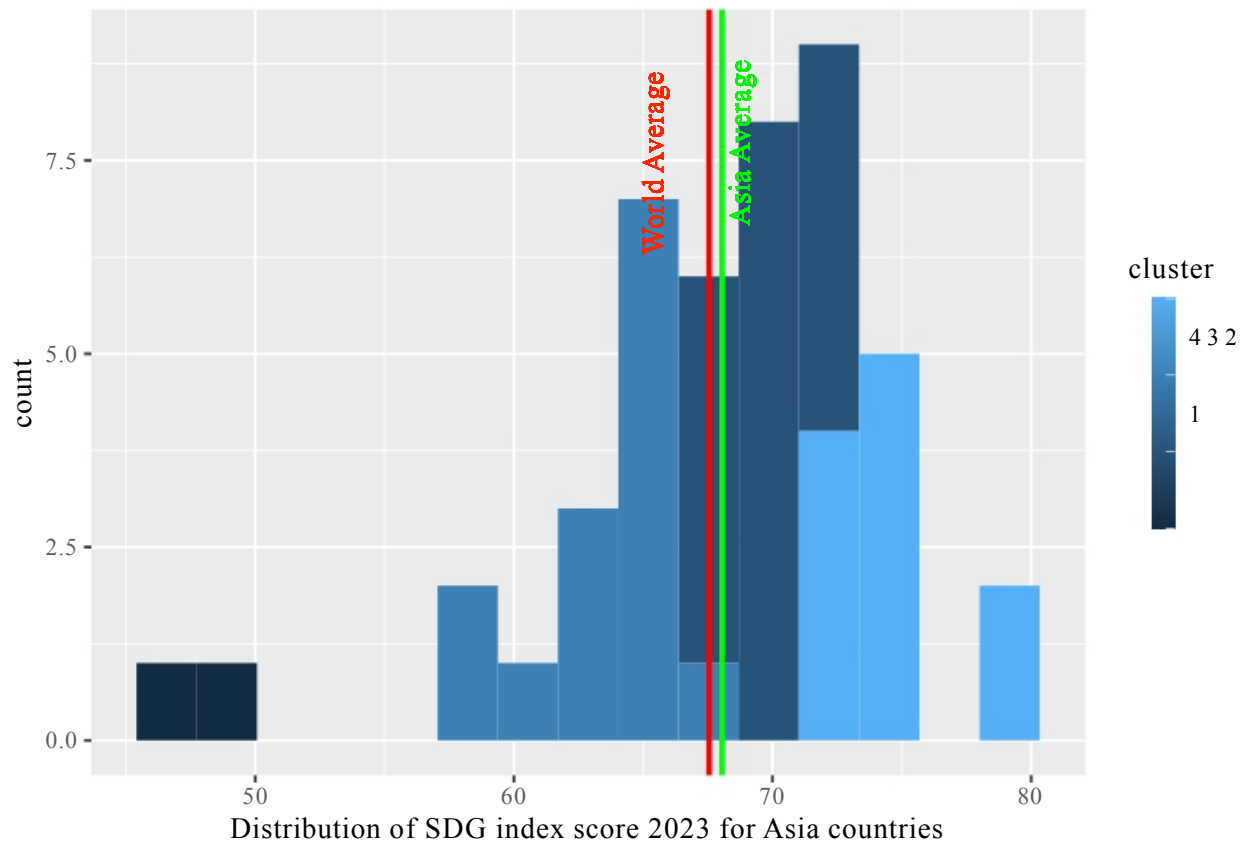
#seperate histogram for each cluster with average mark
y_coordinate_for_label <- 10
clean_data_Asia %>% ggplot(aes(x=SDG_index_score_2023)) +
  geom_histogram(color = "black",
                 fill = "gray",
                 bins = 15) +
  geom_vline(aes(xintercept=world_avg_sdg_index_score), color="red",
             linetype="solid", size=1) +
  geom_text(x=world_avg_sdg_index_score, y=y_coordinate_for_label,
            label="World Average",
            vjust=-1.5, color="red", angle=90, hjust=1.5) +
  #world average index line for better comparison
  geom_vline(aes(xintercept=mean(asia_avg_sdg_index_score)), color="green",
            linetype="solid", size=1) +
  geom_text(x=mean(asia_avg_sdg_index_score), y=y_coordinate_for_label,
            label="Asia Average",
            vjust=2, color="green", angle=90, hjust=1.5) +
  #asia average index line for better comparison
  labs(x="SDG index score 2023",
       y="Number of Countries",
       title="Distribution of SDG index score 2023 for Asia countries") +
  theme(text=element_text(size=18),
        plot.title=element_text(size=14)) +
  facet_wrap(~cluster)

## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

Distribution of SDG index score 2023 for Asia countries

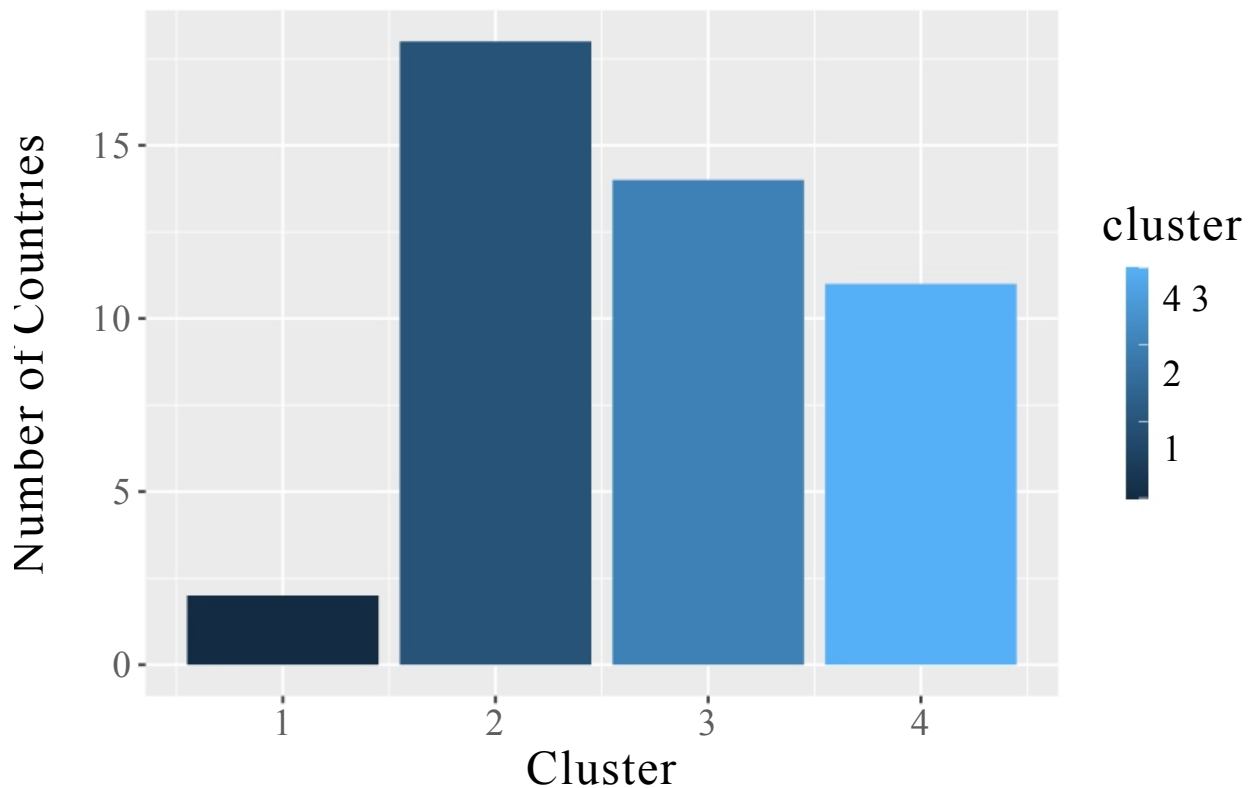


```
#colored histogram with average mark
ggplot(data = clean_data_Asia, aes(x = SDG_index_score_2023,
                                   group = cluster, fill = cluster)) +
  geom_histogram(bins = 15) +
  geom_vline(aes(xintercept=world_avg_sdg_index_score),
             color="red", linetype="solid", size=1) +
  geom_text(x=world_avg_sdg_index_score, y=y_coordinate_for_label,
            label="World Average",
            vjust=-1.5, color="red", angle=90, hjust=1.5) +
#world average index line for better comparison
  geom_vline(aes(xintercept=mean(asia_avg_sdg_index_score)),
             color="green", linetype="solid", size=1) +
  geom_text(x=mean(asia_avg_sdg_index_score), y=y_coordinate_for_label,
            label="Asia Average",
            vjust=2, color="green", angle=90, hjust=1.5) +
#asia average index line for better comparison
  labs(x = "Distribution of SDG index score 2023 for Asia countries")
```



```
# plot number of observations in each cluster
clean_data_Asia %>%
  ggplot() +
  geom_bar(aes(x=cluster, group=cluster, fill=cluster)) +
  labs(x="Cluster",
       y="Number of Countries",
       title="Distribution of SDG index score 2023 for Asia countries") +
  theme(text=element_text(size=18),
        plot.title=element_text(size=14))
```

Distribution of SDG index score 2023 for Asia countries



K-means clustering analyse

<-Clustering enables decision makers to understand complex dataset's pattern directly and easily, so that they can make efficient decisions that optimize resource allocations. (e.g. Each groups that shares the similar pattern can share similar policies)>

<K-mean clustering algorithm is a famous algorithm in statistics that can help us assign data to the corresponding clusters based on minimized variance, which make each groups actually more similar than others. After using it to our data based on SDG Index Score, we managed to split the data into 4 groups.>

<Histogram is a great visualization tool for showing the distribution. A side by side histogram produce by facet~warp by cluster and a histogram colored by group can provide us a more direct and clear view about situation for each cluster and separation between them, which is one of the main purpose of doing clustering compare to the original histogram.>

<We named the first cluster as poorly developed group as it has a average SDG Index Score of 47.90, which is significantly lower than the world average of 67.55 and the Asia average of 68.05. It is marked as the darkest color in the colored histogram and contains 2 countries in it, which occupy a really small percentage compares to other groups, approximately 4.44% of total Asian countries>

<We named the second cluster as upper middle developed group as it has a average SDG Index Score of 69.74, which is higher than the world average of 67.55 and the Asia average of 68.05. It is marked as the second darkest color in the colored histogram and contains 18 countries in it, which takes a huge part out of 47 Asian countries, approximately 40%.>

<We named the third cluster as lower middle developed group as it has a average SDG Index Score of 63.62, which is slightly lower than the world average of 67.55 and the Asia average of 68.05. It is marked as the third darkest color in the colored histogram and contains 14 countries in it, which take up greater part than the highly developed group but smaller than upper middle developed group, at around 31.11% of total Asian countries.>

<We named the fourth cluster as highly developed as it has a average SDG Index Score of 74.59, which is higher than the world average of 67.55, the Asia average of 68.05 and the mean of rest of the clusters. It is marked as the brightest color in the colored histogram and contains 11 countries in it, being the second smallest group, taking up about 24.44%.>

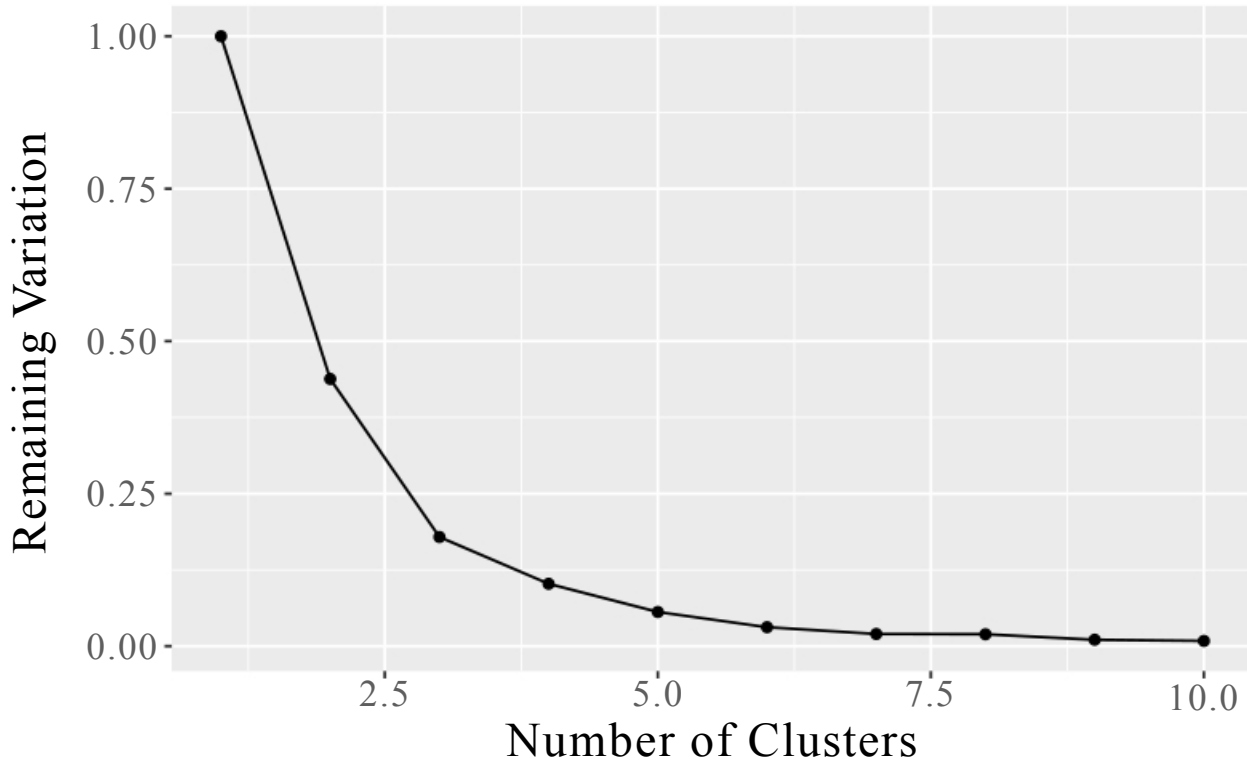
##Conclusion to first part of research question 1

<The distribution of SDG index score is a slightly left skewed normal distribution with one single peak, with most countries' SDG index located near the center. Overall the sustainable development level in Asia is preferable, with a mean higher than the world average, and does not have a huge discrepancies because in spite of the first cluster's mean SDG index is significantly lower(-20.15) than the average level, It actually only occupy 4.44 percent of the Asian countries. The rest 95.56 percent of Asia countries, although being divided into three different groups, does not have significant difference in mean SDG score(63.62, 69.74 and 74.59), mostly located near the center.>

```
#Elbow Approach
k_means_check <- rep(NA, 10)
for(k in 1:10){
  # run k-means on the data
  clustering <- kmeans(clean_data_Asia$SDG_index_score_2023, k)
  k_means_check[k] <- clustering$betweenss / clustering$totss
}

# Plot evolution of metric as a function of k
ggplot() +
  aes(x=1:10, y=1-k_means_check) +
  geom_line() +
  geom_point() +
  labs(x="Number of Clusters",
       y="Remaining Variation",
       title="K-Means Clustering Performance") +
  theme(text=element_text(size=18))
```

K-Means Clustering Performance



Elbow approach for optimal numbers of clusters

<This plot shows us the performance of k-means clustering as we vary the number of clusters from 1 to 10. We see that going from 1 to 3 clusters greatly improves the performance, but the amount of improvement decreases with each additional cluster afterwards. In particular, the performance curve gets quite flat after 4 clusters, so according to elbow approach, using 3 or 4 clusters would be a reasonable choice.>

```
country_groups <- clean_data_Asia %>% select(country_code_iso3,
                                             `Country or Area_en (M49)`, cluster, fsi_total)
country_groups
```

A tibble: 45 x 4

country_code_iso3	`Country or Area_en (M49)`	cluster	fsi_total
<chr>	<chr>	<int>	<dbl>
1 AFG	Afghanistan	1	107.
2 ARM	Armenia	4	67.5
3 AZE	Azerbaijan	4	72.7
4 BHR	Bahrain	3	65.1
5 BGD	Bangladesh	3	85.2
6 BTN	Bhutan	4	66.4
7 BRN	Brunei Darussalam	3	54.7
8 KHM	Cambodia	3	80.3
9 CHN	China	2	65.1
10 CYP	Cyprus	4	57

i 35 more rows

#summary table group by cluster

```
country_groups %>% group_by(cluster) %>%
  summarise(
```



```
n = n(),
cluster_fsi_mean = mean(fsi_total),
countries = toString(`Country or Area_en (M49)`)
)
```

```
## # A tibble: 4 x 4
```

```
##   cluster      n cluster_fsi_mean countries
##   <int> <int>      <dbl> <chr>
## 1     1      2      108. Afghanistan, Yemen
## 2     2     18      66.3 China, Indonesia, Iran (Islamic Republic of), ~
## 3     3     14      74.7 Bahrain, Bangladesh, Brunei Darussalam, Cambod~
## 4     4     11      58.5 Armenia, Azerbaijan, Bhutan, Cyprus, Georgia, ~
```

#Does It make sense for our k-means clustering when we combine our result with other variables
#How is fsi index fit and related to the clustering we made above

```
clean_data_Asia %>% ggplot(aes(x=SDG_index_score_2023, y=fsi_total, color = cluster)) +
  geom_point() +
  labs(x = "SDG Index Score",
       y = "FSI Score")
```



```
## Conclusion to second part of research question 1
```

<From our scatter plot we can see that the first cluster(Darkest color), the poorly developed group, located at the top left corner, where the FSI Score is highest and SDG Score is lowest, which make quite sense as FSI score evaluate a country's stability, the higher the score is, the lower the stability is. A poor stability will likely to not being beneficial to sustainable development. The second cluster(second darkest color) has a FSI Score mean of 66.3, which is lower than 74.7 of cluster 3(third darkest color) who has a higher lower mean SDG Score. It is also higher than the FSI Score mean 58.5 of cluster 4, who has the highest average SDG score. These also make sense as more stable the country is, the more likely it is going to have a better development

environment and get a higher sustainable development score. Both FSI Score and SDG score data comes from reliable non-profitable groups, So overall our cluster for sustainable development level will generally make sense. However, it's essential to note that while there might be a general trend, the correlation does not necessarily imply causation, and outliers or confoundings might exist due to specific national circumstances or other unmeasured factors.>

##Research Question 2 Part

#rename long name variables

```
final_data <- final_data %>%
  rename(income_share = `sowc_social-protection-and-equity__share-of-household-income-2010-2019-
    r_bottom-40` )
```

#Set seed

```
set.seed(1009891468)
```

#Data cleaning

```
data_Western_Q2_cleaned <- final_data %>%
  filter(`Region Code (M49)`==150 | `Region Code (M49)`==19) %>%
  distinct(country_code_iso3, .keep_all = TRUE) %>% #drop dup
  filter(!
    is.na(income_share) & income_share != Inf & income_share != -Inf)
```

```
data_Asia_Q2_cleaned <- final_data %>% filter(`Region Code (M49)`==142) %>%
  distinct(country_code_iso3, .keep_all = TRUE) %>% #drop dup
  filter(!
    is.na(income_share) & income_share != Inf & income_share != -Inf)
```

#filter out the variable of share of household income

```
Asia_income <- data_Asia_Q2_cleaned %>%
  select(income_share) %>%
  mutate(region = 'Asia')
```

```
Western_income <- data_Western_Q2_cleaned %>%
  select(income_share) %>%
  mutate(region = 'Western')
```

#Join tables

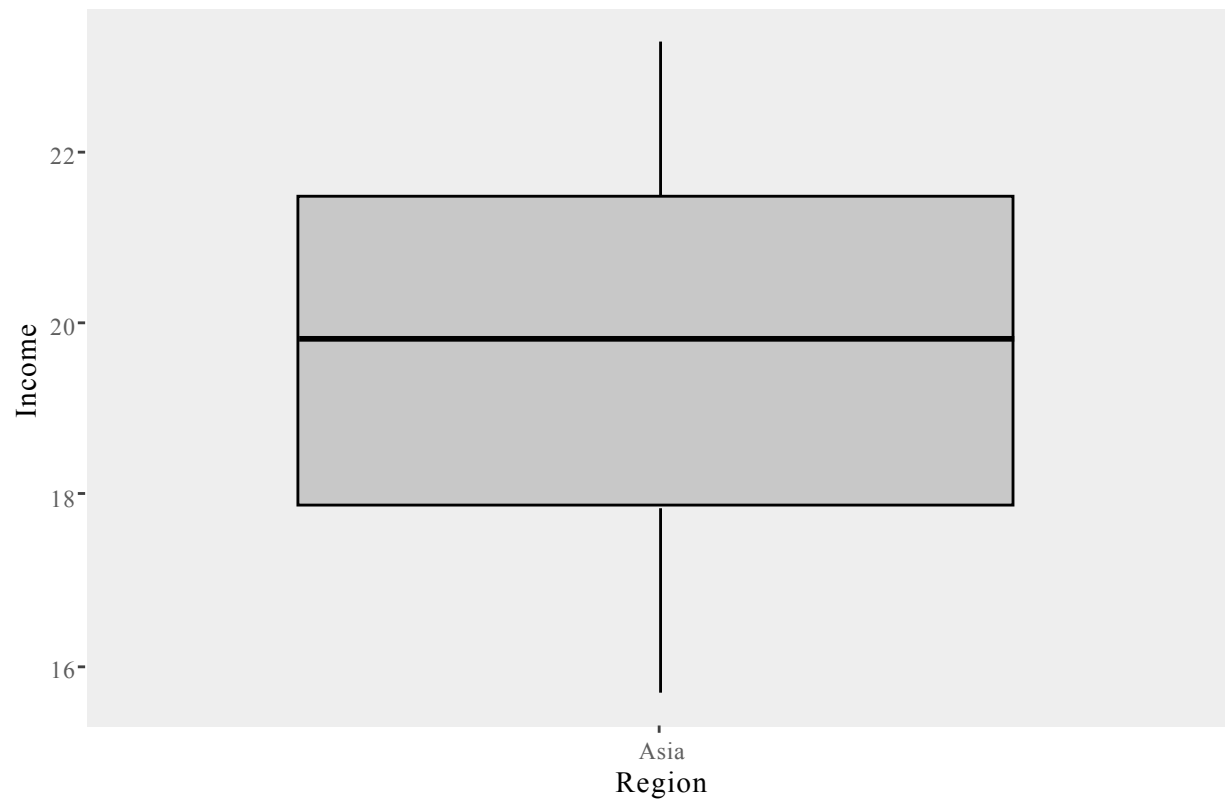
```
Asia_Western <- full_join(x = Asia_income, y = Western_income)
```

Joining with `by = join_by(income_share, region)`

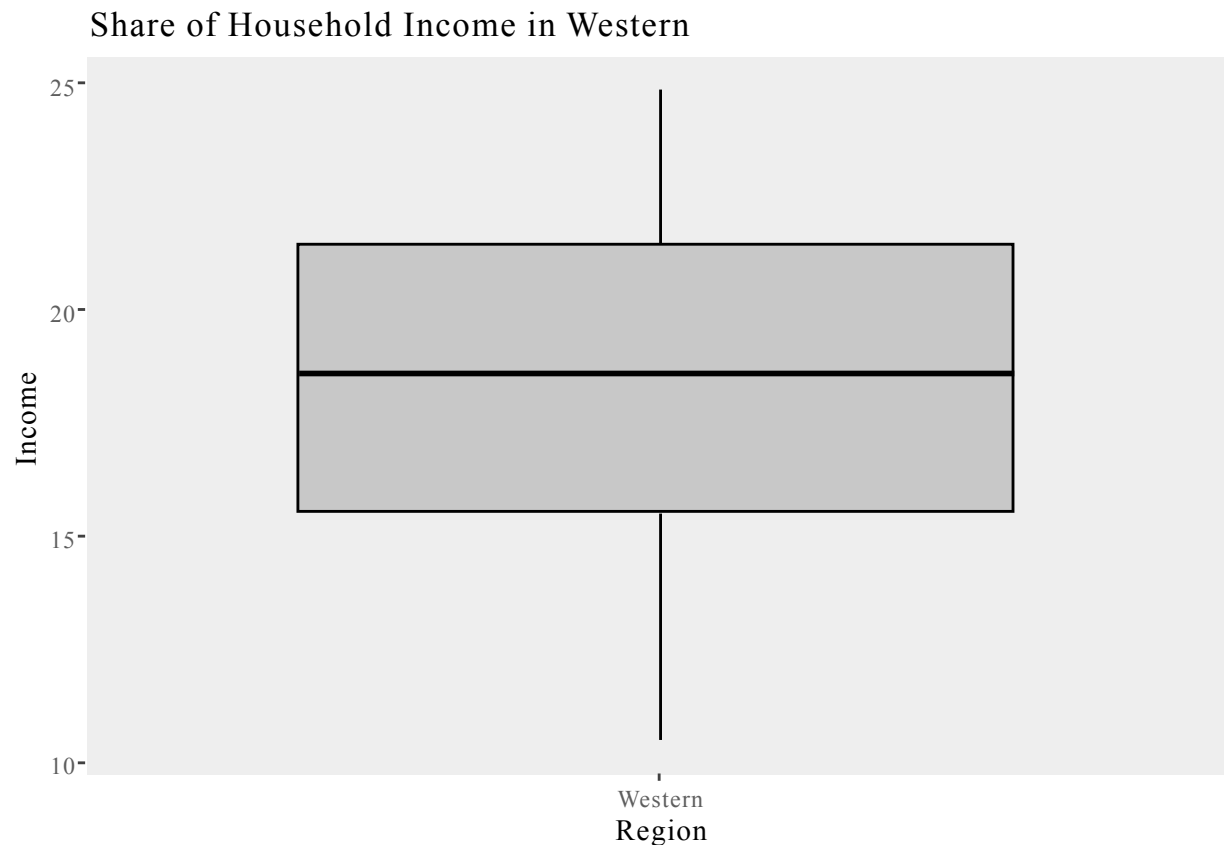
#Make the visualizations of Asia and Western's share of household income

```
ggplot(data = Asia_income,
  aes(x = region,
    y = income_share)) +
  geom_boxplot(color = 'black', fill = 'grey') +
  labs(x = "Region", y = "Income", title = "Share of Household Income in Asia")
```

Share of Household Income in Asia



```
ggplot(data = Western_income,  
       aes(x = region,  
           y = income_share)) +  
  geom_boxplot(color = 'black', fill = 'grey') +  
  labs(x = "Region", y = "Income", title = "Share of Household Income in Western")
```



```
#Two-Sample Hypothesis Test
#Null Hypothesis: There is no difference between Asia and Western Countries.
#Alternative: There is difference between them.

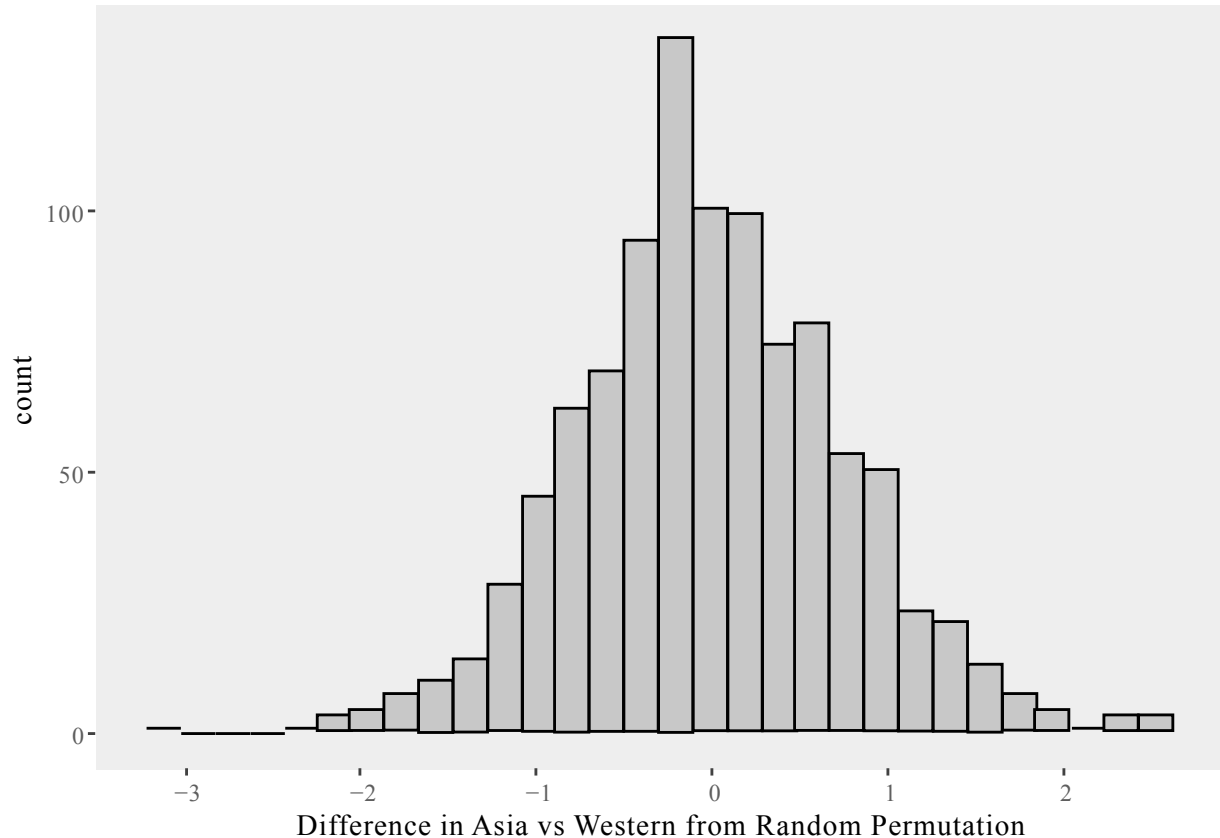
#Calculate the test statistic
Asia_mean <- mean(data_Asia_Q2_cleaned$
  income_share,
  na.rm = TRUE)

Western_mean <- mean(data_Western_Q2_cleaned$
  income_share,
  na.rm = TRUE)

observed_test_statistic <- Asia_mean - Western_mean

#Random Permutation Test
num_trials <- 1000
delta_mean_simulations <- rep(NA, num_trials)
for (i in 1:num_trials) {
  sim_data <- Asia_Western %>% mutate(region = sample(region, replace = FALSE))
  delta_mean_sim <- sim_data %>% group_by(region) %>%
    summarise(means = mean(income_share,
      na.rm = TRUE), .groups='drop') %>%
    summarise(value = diff(means)) %>%
    as.numeric()
  delta_mean_simulations[i] <- delta_mean_sim
}
```

```
#Make the visulization of the permutation test
ggplot() + aes(x = delta_mean_simulations) +
  geom_histogram(color = "black", fill = "gray", bins = 30) +
  labs(x = 'Difference in Asia vs Western from Random Permutation')
```



```
#Calculate P-value
p_value <- sum(abs(delta_mean_simulations) >= abs(observed_test_statistic)) / num_trials
```

#Conclusion for research question 2 <The p-value calculated here is 0.117, which is higher than the alpha level 0.05. Thus, the null hypothesis would be accepted, which states that there is no difference between the household income in Asia and Western. This conclusion can also be reflected by the boxplots that are made at the beginning, which shows that the mean income between the continents are similar. However, the data set contains too many non-values and the household income may not represent the whole economic growth of the continents. Hence, there may result in a type II error where it fails to reject the hypothesis. A more specific analysis needs larger samples and more concise values. >

##Research Question 3 Part

```
#rename too long variables and data cleaning
final_data <- final_data %>%
  rename(mortality_rate = `sowc_child-mortality__infant-mortality-rate_2021`) %>%
  rename(growth_rate = `sowc_demographics__annual-population-growth-rate_2000-2020`)
clean_data_Asia_nick <- data_Asia %>%
  rename(mortality_rate = `sowc_child-mortality__infant-mortality-rate_2021`) %>%
  rename(growth_rate = `sowc_demographics__annual-population-growth-rate_2000-2020`) %>%
  distinct(country_code_iso3, .keep_all = TRUE) %>% #drop dup
  filter(!is.na(mortality_rate) &
    mortality_rate !=
```

```

      Inf & mortality_rate != -Inf)
final_data_clean <- final_data %>%
distinct(country_code_iso3, .keep_all = TRUE) %>% #drop dup
filter(!is.na(mortality_rate) &
       mortality_rate != Inf &
       mortality_rate != -Inf)

nick_asia_tible <- clean_data_Asia_nick %>%
select(mortality_rate, growth_rate)
glimpse(nick_asia_tible)

## Rows: 46
## Columns: 2
## $ mortality_rate <dbl> 43.391630, 9.544795, 16.606150, 5.913218, 22.906148, 22~
## $ growth_rate      <dbl> 3.28682421, 0.57926313, 1.08441292, 3.47996129, 1.23429~
#set seed
set.seed(1009853874)

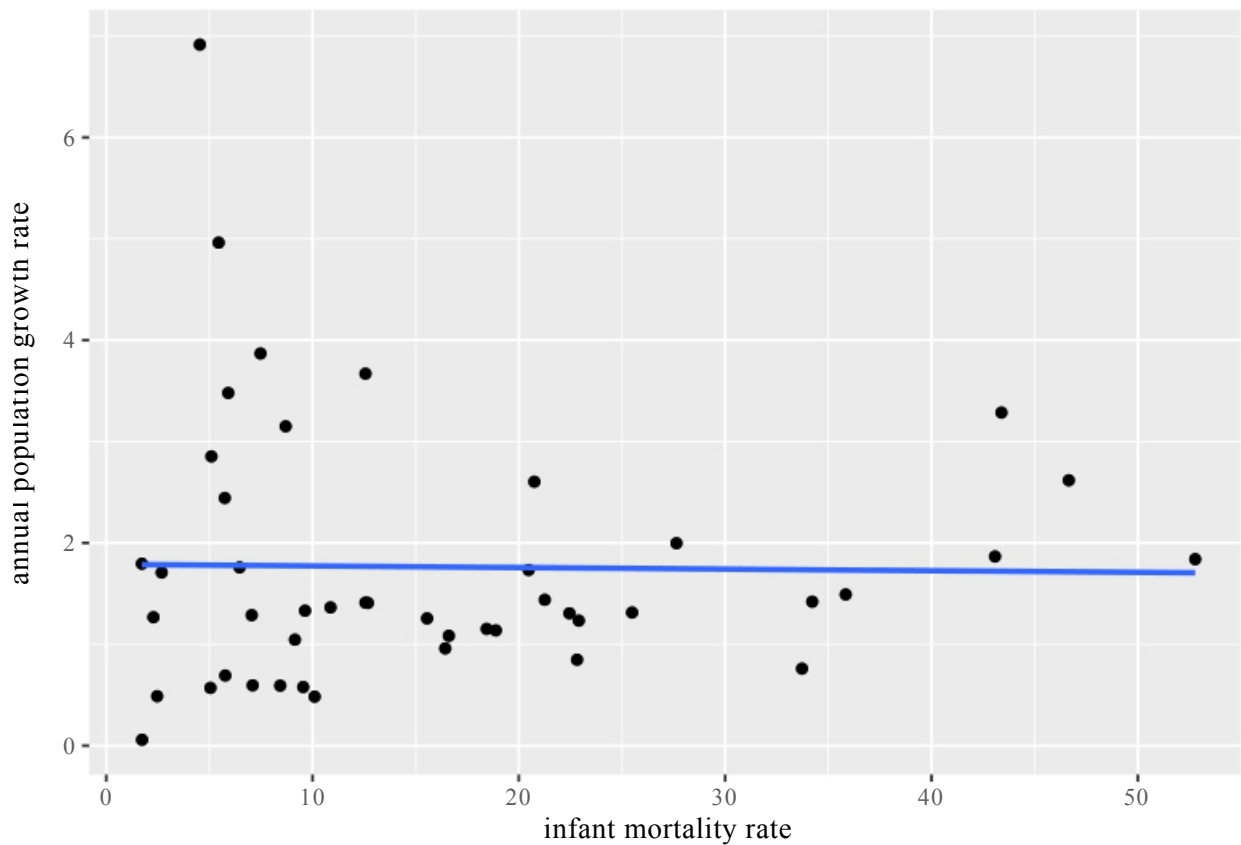
#linear regression single line
clean_data_Asia_nick %>% ggplot(aes(x=mortality_rate,
                                   y=growth_rate)) +

  geom_point() +
  labs(x = "infant mortality rate",
       y = "annual population growth rate") +

  geom_point(alpha=0.5) + geom_smooth(method = "lm", se=FALSE)

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

```



```
nick_regression_model <- lm(mortality_rate ~
  growth_rate,
  data = clean_data_Asia_nick)
summary(nick_regression_model)$coefficients
```

```
##           Estimate Std. Error   t value    Pr(>|t|)
## (Intercept) 16.3753178   3.324725  4.9253156 1.233442e-05
## growth_rate -0.1613022   1.527998 -0.1055644 9.164078e-01
```

#other two linear regression line

```
new_regression_multi_data <- clean_data_Asia_nick %>%
  select(mortality_rate,
    growth_rate)
```

#make new regression for high mortality rate

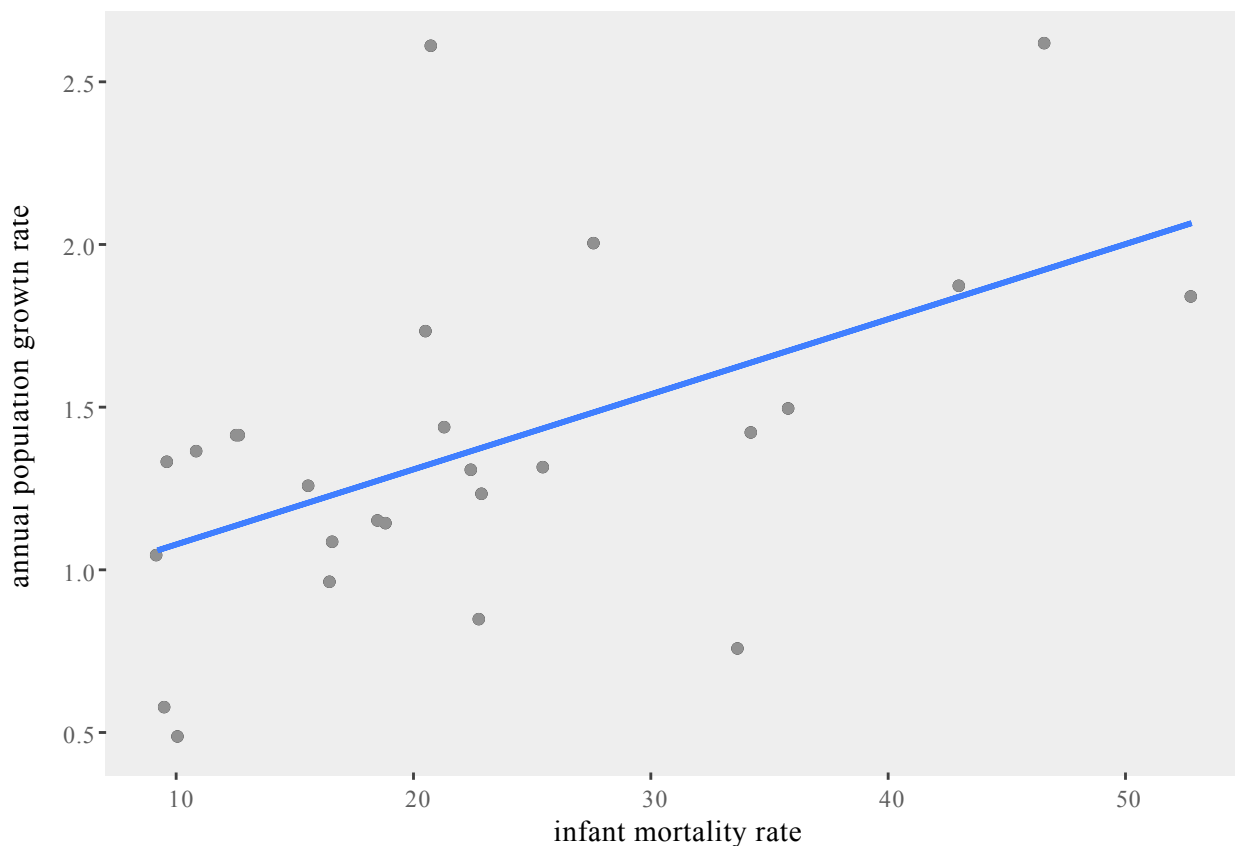
```
regression_high_mortality <- new_regression_multi_data %>%
  filter(mortality_rate >= 9)%>%
  filter(growth_rate <= 3)
regression_high_mortality
```

```
## # A tibble: 26 x 2
##   mortality_rate growth_rate
##   <dbl>         <dbl>
## 1      9.54      0.579
## 2     16.6      1.08
## 3     22.9      1.23
## 4     22.5      1.31
## 5      9.64      1.33
```

```
## 6      21.3      1.44
## 7      10.1      0.484
## 8      25.5      1.31
## 9      18.9      1.14
## 10     10.9      1.36
## # i 16 more rows
```

```
regression_high_mortality %>% ggplot(aes(x=mortality_rate,
                                           y=growth_rate )) +
  geom_point() +
  labs(x = "infant mortality rate",
       y = "annual population growth rate") + geom_point(alpha=0.5) +
  geom_smooth(method = "lm", se=FALSE)
```

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



```
nick_high_regression_model <- lm(mortality_rate
                                ~ growth_rate,
                                data = regression_high_mortality)
summary(nick_high_regression_model)$coefficients
```

```
##              Estimate Std. Error  t value Pr(>|t|)
## (Intercept)  5.819305    5.843085  0.9959304 0.3292199
## growth_rate 12.305345    3.993713  3.0811790 0.0051131
```

```
#make new regression for low mortality rate
regression_low_mortality <- new_regression_muulti_data %>%
  filter(mortality_rate <= 10)%>%
  filter(growth_rate>=0)
```

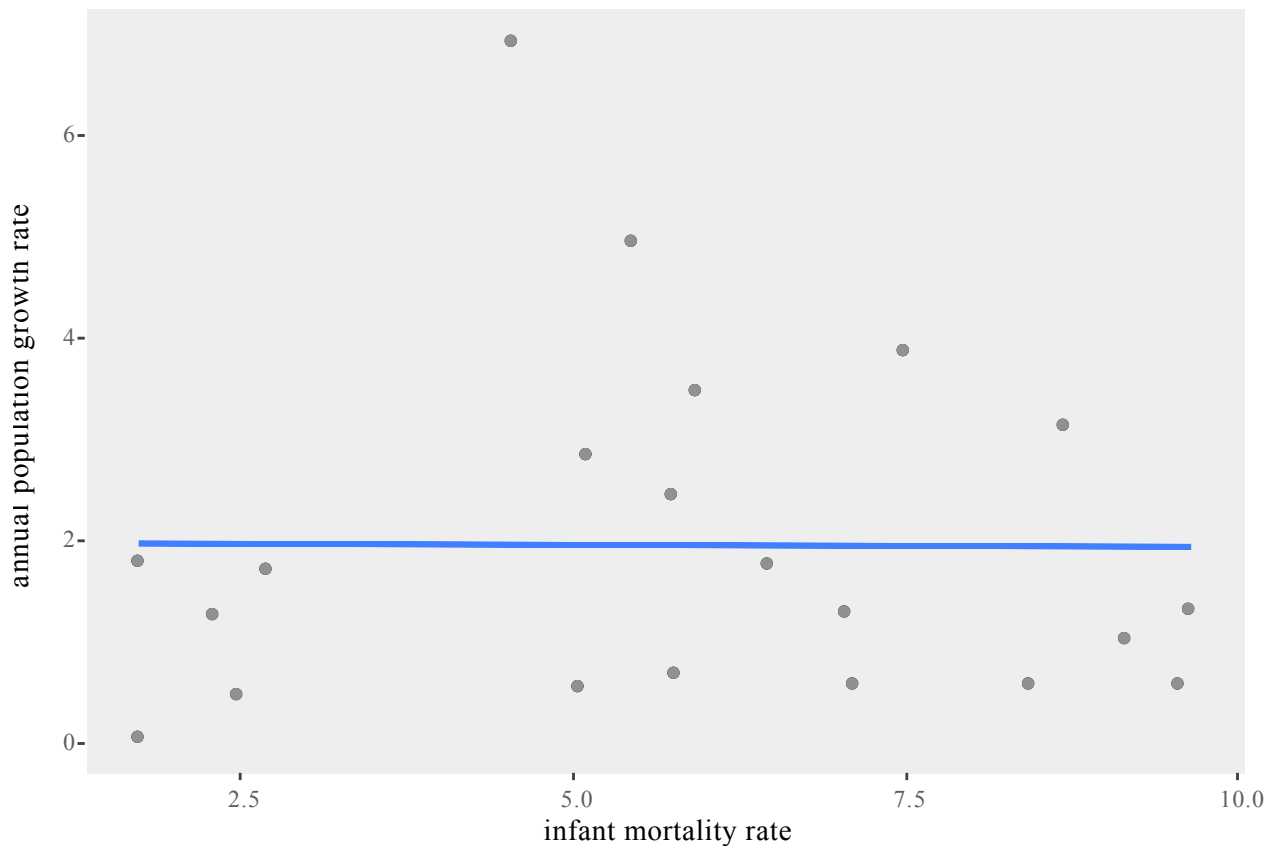


```
regression_high_mortality
```

```
## # A tibble: 26 x 2
##   mortality_rate growth_rate
##   <dbl>         <dbl>
## 1      9.54      0.579
## 2     16.6      1.08
## 3     22.9      1.23
## 4     22.5      1.31
## 5      9.64      1.33
## 6     21.3      1.44
## 7     10.1      0.484
## 8     25.5      1.31
## 9     18.9      1.14
## 10    10.9      1.36
## # i 16 more rows
```

```
regression_low_mortality %>% ggplot(aes(x=mortality_rate,
                                          y=growth_rate )) +
  geom_point() +
  labs(x = "infant mortality rate",
       y = "annual population growth rate") + geom_point(alpha=0.5) +
  geom_smooth(method = "lm", se=FALSE)
```

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



```
nick_low_regression_model <- lm(mortality_rate ~ growth_rate,
                                data = regression_low_mortality)
```

```
summary(nick_low_regression_model)$coefficients
```

```
##              Estimate Std. Error    t value    Pr(>|t|)
## (Intercept)  5.828040391  0.8833906  6.59735391 2.585314e-06
## growth_rate -0.009085322  0.3408511 -0.02665481 9.790130e-01
```

#Conclusion to research question 3 #Conclusion <We suspect countries with higher mortality rate also have higher population growth rate, and lower mortality rate also have lower mortality rate. We also suspect there is a certain percentage below such that those countries population growth rate would not be related to the infant mortality rate.> <In the scatter graph we could see some extreme values, there are two countries that stand out the most, one of them is having highest population growth rate with low mortality rate, and the other one is quite the opposite, it has the highest infant mortality rate but has lower annual population rate.> <From what we get from the regression line, we can see the slope of the line is almost zero, which means there is no relation between infant mortality rate and population growth rate in Asia. But when we look closely, we can tell that there are two trends that cancel each other out. One is from lower annual population rate and higher population growth rate, other is the opposite. As a result, I will try to make two regression lines.> <For the higher mortality rate countries, there is a clear upward trend, shows how for those countries who have relatively higher mortality rate will likely to have higher annual population growth rate. Which is accurate because we suspect countries with higher mortality rate also have higher population growth rate.> <For lower mortality rate countries, the regression line is flat. Which means that there is no clear trend between infant mortality rate and population growth rate. Which is also intuitive, because we can see under 10 infant mortality rate we can see the regression is flat.> #Limitations and Concerns <First of the limitation and concerns we see is that linear regression is sensitive to outliers, which means some of the extreme values would affect our prediction. Second of the limitations is that we only looked at the effect of infant mortality rate, which is only a small part of country population growth. There are other factors affect the outcome.>

*#Reference Page <- "Remove NA Rows in R." ProgrammingR, <https://www.programmingr.com/examples/remove-na-rows-in-r/>. 2024> <- "United Nations Development Programme." Sustainable Development Goals, United Nations Development Programme, <https://www.undp.org/sustainable-development-goals>. , Updated in 2023> <- "Fragile States Index." The Fund for Peace, <https://fragilestatesindex.org/>. Updated in 2023> <- "Statistical Tables: The State of the World's Children 2023." UNICEF Data, UNICEF, <https://data.unicef.org/resources/dataset/the-state-of-the-worlds-children-2023-statistical-tables/>. Updated in 2023> <- ChatGPT 3.5 was used in order to summarize and check grammar and sentence structure for conclusion. <https://chat.openai.com/share/c0249b58-d0b1-4731-909b-b9ae6ba5c76f>> <- We reference to the code style of Class Examples and Weekly Problem Sets, we also ask for help from TAs.> >**