# Country data for donation priority ranking

About the Dataset and its Organization:

HELP International is a global humanitarian NGO dedicated to combating poverty and providing essential amenities and relief to the people in underdeveloped countries, especially during disasters and natural calamities.

Overview of the Dataset:

The dataset utilized in this project was sourced from a public Kaggle profile. It encompasses information on 9 socio-economic and health factors for 167 countries:

- country: Name of the country
- child_mort: Child mortality rate, indicating the number of deaths of children under 5 years per 1000 live births
- exports: Exports of goods and services per capita, presented as a percentage of the GDP per capita
- health: Total health spending per capita, represented as a percentage of GDP per capita
- imports: Imports of goods and services per capita, expressed as a percentage of the GDP per capita
- Income: Net income per person
- Inflation: Annual growth rate measurement of the Total GDP
- life_expec: Average number of years a newborn child would live if the current mortality patterns persist
- total_fer: Number of children that would be born to each woman if the current age-fertility rates remain constant
- gdpp: GDP per capita, calculated as the Total GDP divided by the total population.

```
country <- read.csv("Country-data.csv")
#str(country)
```

1. Filtering missing values and classyfing data type

```
summary(is.na(country))
```

```
##   country        child_mort       exports          health
##  Mode :logical   Mode :logical   Mode :logical   Mode :logical
##  FALSE:167       FALSE:167       FALSE:167       FALSE:167
##   imports          income         inflation       life_expec
##  Mode :logical   Mode :logical   Mode :logical   Mode :logical
##  FALSE:167       FALSE:167       FALSE:167       FALSE:167
##  total_fer          gdpp
##  Mode :logical   Mode :logical
##  FALSE:167       FALSE:167
```

Upon inspecting for missing values using the "summary(is.na())" function, it was determined that the dataset contains no missing values(result for missing values are FALSE).

The current data types for each column are considered appropriate. However, there is an option to change the data type of the "country" column to a factor if desired. Therefore, the next step involves proceeding with the conversion of the data type for the "country" column.

2. Exact Problem setting The primary goal is to classify countries based on their socio-economic and health factors, determining their overall level of development. As part of this process, the data type of the "country" column will be changed to a factor. This classification aims to facilitate the selection of a country for monetary donations based on its development status.

```
country <- country |> mutate(country = as.factor(country))
```

3. Explore the pattern of data

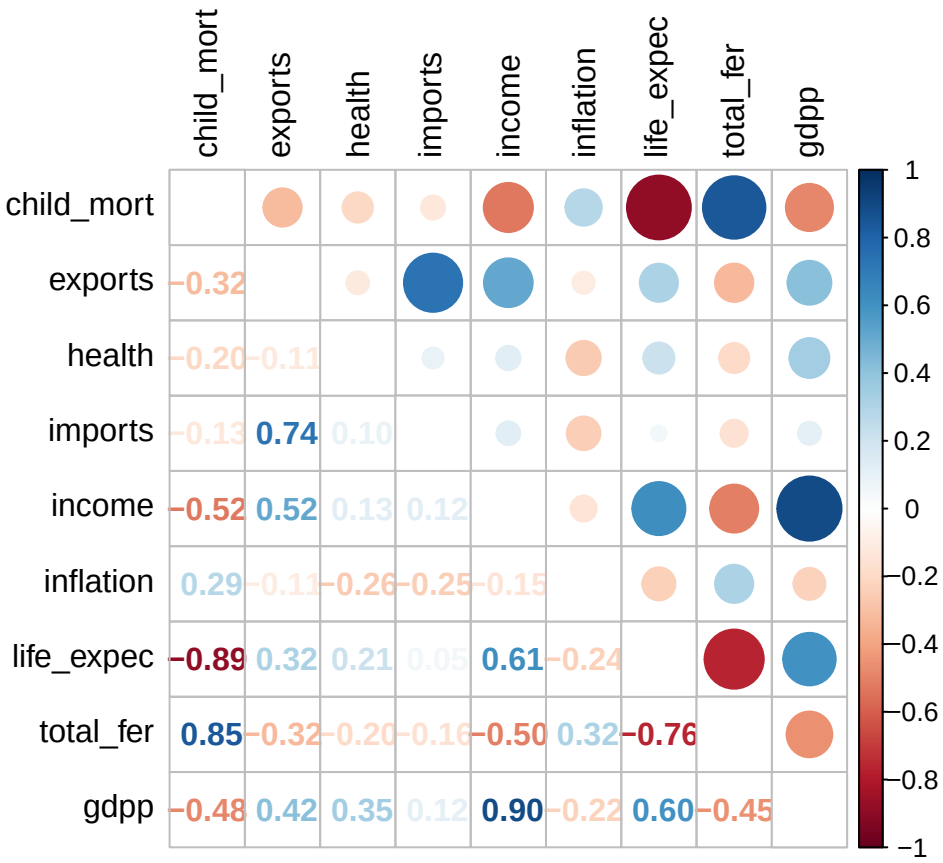
```r
summary(country)
```

```
##                 country      child_mort        exports           health
##  Afghanistan        :  1   Min.   :  2.60   Min.   :  0.109   Min.   : 1.810
##  Albania            :  1   1st Qu.:  8.25   1st Qu.: 23.800   1st Qu.: 4.920
##  Algeria            :  1   Median : 19.30   Median : 35.000   Median : 6.320
##  Angola             :  1   Mean   : 38.27   Mean   : 41.109   Mean   : 6.816
##  Antigua and Barbuda:  1   3rd Qu.: 62.10   3rd Qu.: 51.350   3rd Qu.: 8.600
##  Argentina          :  1   Max.   :208.00   Max.   :200.000   Max.   :17.900
##  (Other)            :161
##     imports             income         inflation         life_expec
##  Min.   :  0.0659   Min.   :   609   Min.   : -4.210   Min.   :32.10
##  1st Qu.: 30.2000   1st Qu.:  3355   1st Qu.:  1.810   1st Qu.:65.30
##  Median : 43.3000   Median :  9960   Median :  5.390   Median :73.10
##  Mean   : 46.8902   Mean   : 17145   Mean   :  7.782   Mean   :70.56
##  3rd Qu.: 58.7500   3rd Qu.: 22800   3rd Qu.: 10.750   3rd Qu.:76.80
##  Max.   :174.0000   Max.   :125000   Max.   :104.000   Max.   :82.80
##
##    total_fer          gdpp
##  Min.   :1.150   Min.   :   231
##  1st Qu.:1.795   1st Qu.:  1330
##  Median :2.410   Median :  4660
##  Mean   :2.948   Mean   : 12964
##  3rd Qu.:3.880   3rd Qu.: 14050
##  Max.   :7.490   Max.   :105000
##
```

```r
Vars_with_num <- which(map(country, is.numeric) == T) #Compute var.

cat('There are', length(Vars_with_num), 'Numeric Variables from columns')
```

```
## There are 9 Numeric Variables from columns
```

```r
allVar <- country[, Vars_with_num]
#For non-NA values, pairwise correlation.
num_Var <- cor(allVar, use="pairwise.complete.obs")


corrplot.mixed(num_Var, tl.col = "black", tl.pos = "lt")
```



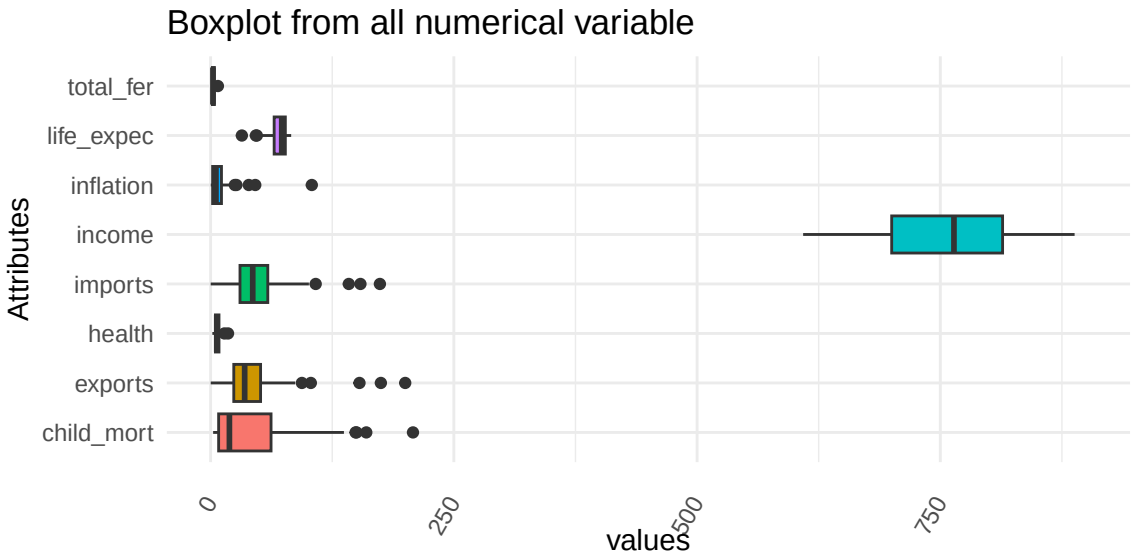

The data distribution depicted above unveils a notable disparity in the quality of life among countries, presenting a poignant scenario. Additionally, following the conducted inspection, it becomes imperative to normalize the data due to the distinct value ranges in the columns. Standardization is deemed essential to mitigate bias arising from disparate variance or covariance values.

The correlation matrix serves as an initial tool for comprehending the interplay between variables and discerning potential relationships within final clusters. The outcomes highlight that the income/gdpp pair exhibits the highest correlation, succeeded by child_mort/total_fer and exports/imports. These correlations are sensible indicators for segmenting countries into clusters based on their overall development.

Nevertheless, our objective is to encompass all variables within the dataset and classify the data accordingly. Therefore, the subsequent pivotal step involves dimension reduction.

```
country |>
  gather(Attributes, values, c(2:9)) |>
  ggplot(aes(y=values, x=Attributes,  fill=Attributes)) +
  ylim(0, 900) +
  geom_boxplot(show.legend=FALSE)  +
  coord_flip()+

  labs(title="Boxplot from all numerical variable") +
  theme_minimal()+ theme(axis.text.x = element_text(angle = 60, vjust = 0.5, hjust=1))
```

```
## Warning: Removed 170 rows containing non-finite values (‘stat_boxplot()‘).
```




Despite the presence of outliers in several variables, the decision has been made to retain them. These outliers potentially represent countries facing extremely dire circumstances and may qualify for financial assistance.

Moving on to the next step:

4. Rescaling Data

Given that the first column in the dataset is of string type, it is necessary to exclude it before applying the dataset to the `scale()` function. Including any numeric data type in the `scale()` function would lead to an error.

```
scale <- country |> mutate_at(c(2:10), funs(c(scale(.))))
```

```
## Warning: `funs()` was deprecated in dplyr 0.8.0.
## i Please use a list of either functions or lambdas:
##
## # Simple named list: list(mean = mean, median = median)
##
## # Auto named with `tibble::lst()`: tibble::lst(mean, median)
##
## # Using lambdas list(~ mean(., trim = .2), ~ median(., na.rm = TRUE))
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

```
head(scale)
```

```
##               country child_mort     exports      health     imports
## 1         Afghanistan  1.2876597 -1.13486665  0.27825140 -0.08220771
## 2             Albania -0.5373329 -0.47822017 -0.09672528  0.07062429
## 3             Algeria -0.2720146 -0.09882442 -0.96317624 -0.63983800
## 4              Angola  2.0017872  0.77305618 -1.44372888 -0.16481961
## 5 Antigua and Barbuda -0.6935483  0.16018613 -0.28603389  0.49607554
## 6           Argentina -0.5894047 -0.81019144  0.46756001 -1.27594958
##        income  inflation life_expec   total_fer        gdpp
## 1 -0.80582187  0.1568645 -1.6142372  1.89717646 -0.67714308
## 2 -0.37424335 -0.3114109  0.6459238 -0.85739418 -0.48416709
## 3 -0.22018227  0.7869076  0.6684130 -0.03828924 -0.46398018
## 4 -0.58328920  1.3828944 -1.1756985  2.12176975 -0.51472026
## 5  0.10142673 -0.5999442  0.7021467 -0.54032130 -0.04169175
## 6  0.08067776  1.2409928  0.5897009 -0.38178486 -0.14535428
```

After comparing the summary results of the “country” object and the “scale” object, it is evident that there is no significant difference in the range of values for each column in the “scale” object. The data in each column exhibits the same minimum and maximum range of values, providing improved results during the machine learning modeling process.

5. Principal Component Analysis

Principal Component Analysis (PCA) is a widely used dimensionality reduction technique that decreases the dimensionality of large datasets by transforming a large set of variables into a smaller set that retains most of the information from the larger set.

Reducing variables in a dataset may lead to a slight decrease in accuracy, but the main advantage of dimensionality reduction is that it trades a small amount of accuracy for simplicity. Smaller datasets are simpler to explore and visualize, making it easier and faster for machine learning algorithms to process without unnecessary variables.

5.1 Basic Computing for PCA

In R, PCA can be performed using the `prcomp()` function. However, implementing the `prcomp()` function within the `mutate_at()` function is not possible. Therefore, numeric values must be separated before conducting PCA.

```
rownames(scale) <- scale[,"country"]

scale <- scale %>% select(-country)

head(scale)
```

```
##                     child_mort     exports      health     imports      income
## Afghanistan          1.2876597 -1.13486665  0.27825140 -0.08220771 -0.80582187
## Albania             -0.5373329 -0.47822017 -0.09672528  0.07062429 -0.37424335
## Algeria             -0.2720146 -0.09882442 -0.96317624 -0.63983800 -0.22018227
## Angola               2.0017872  0.77305618 -1.44372888 -0.16481961 -0.58328920
## Antigua and Barbuda -0.6935483  0.16018613 -0.28603389  0.49607554  0.10142673
## Argentina           -0.5894047 -0.81019144  0.46756001 -1.27594958  0.08067776
##                      inflation life_expec   total_fer        gdpp
## Afghanistan          0.1568645 -1.6142372  1.89717646 -0.67714308
## Albania             -0.3114109  0.6459238 -0.85739418 -0.48416709
## Algeria              0.7869076  0.6684130 -0.03828924 -0.46398018
## Angola               1.3828944 -1.1756985  2.12176975 -0.51472026
## Antigua and Barbuda -0.5999442  0.7021467 -0.54032130 -0.04169175
## Argentina            1.2409928  0.5897009 -0.38178486 -0.14535428
```

```
PCAcountry <- prcomp(scale)
PCAcountry
```

```
## Standard deviations (1, .., p=9):
## [1] 2.0336314 1.2435217 1.0818425 0.9973889 0.8127847 0.4728437 0.3368067
## [8] 0.2971790 0.2586020
##
## Rotation (n x k) = (9 x 9):
##                    PC1          PC2         PC3          PC4         PC5
## child_mort -0.4195194 -0.192883937  0.02954353  0.370653262 -0.16896968
## exports     0.2838970 -0.613163494 -0.14476069  0.003091019  0.05761584
## health      0.1508378  0.243086779  0.59663237  0.461897497  0.51800037
## imports     0.1614824 -0.671820644  0.29992674 -0.071907461  0.25537642
## income      0.3984411 -0.022535530 -0.30154750  0.392159039 -0.24714960
## inflation  -0.1931729  0.008404473 -0.64251951  0.150441762  0.71486910
## life_expec  0.4258394  0.222706743 -0.11391854 -0.203797235  0.10821980
## total_fer  -0.4037290 -0.155233106 -0.01954925  0.378303645 -0.13526221
## gdpp        0.3926448  0.046022396 -0.12297749  0.531994575 -0.18016662
##                     PC6         PC7         PC8         PC9
## child_mort -0.200628153 -0.07948854  0.68274306 -0.32754180
## exports     0.059332832 -0.70730269  0.01419742  0.12308207
## health     -0.007276456 -0.24983051 -0.07249683 -0.11308797
## imports     0.030031537  0.59218953  0.02894642 -0.09903717
## income     -0.160346990  0.09556237 -0.35262369 -0.61298247
## inflation  -0.066285372  0.10463252  0.01153775  0.02523614
## life_expec  0.601126516  0.01848639  0.50466425 -0.29403981
## total_fer   0.750688748  0.02882643 -0.29335267  0.02633585
## gdpp       -0.016778761  0.24299776  0.24969636  0.62564572
```

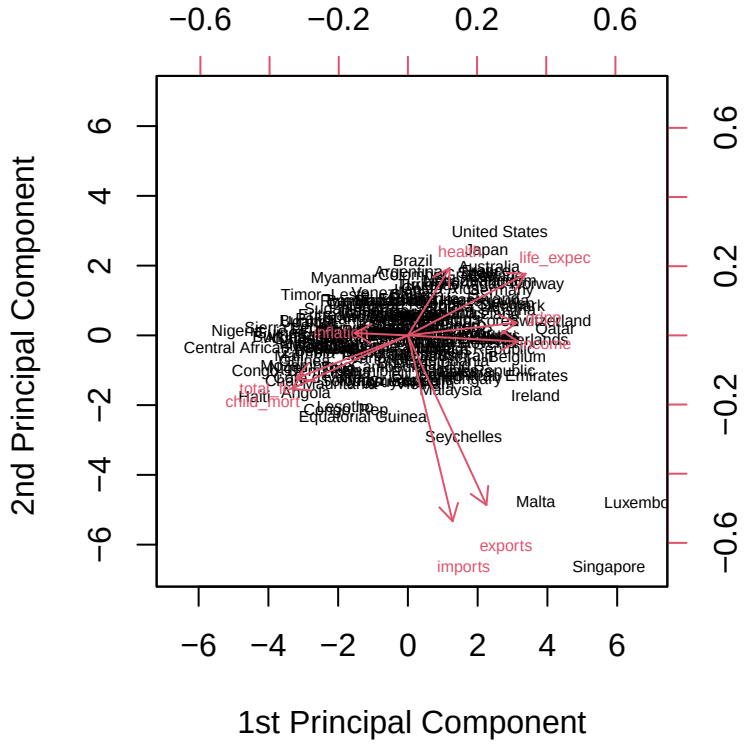
Interpretation of PCA Components:

The PCA components can be interpreted in multiple ways.

1. Principal Component 1 (PC1):
   - Positively associated with: Exports, health, imports, income, life expectancy, and GDP per capita (gdpp).
   - Negatively associated with: Child mortality (child_mort), inflation, and total fertility rate (total_fer).
   - Interpretation: PC1 can be viewed as a measure of a country's stability. A high score on PC1 indicates higher stability, characterized by positive attributes such as high income, life expectancy, and GDP, and lower values in negative attributes like child mortality, inflation, and total fertility rate.
2. Coefficients and Variable Importance:
   - The absolute values of coefficients indicate the importance of variables in calculating each component.
   - Example: In PC7, "exports" has the largest coefficient, suggesting that PC7 heavily relies on information from the "exports" variable.
3. Biplot Visualization:
   - The biplot() function is useful for observing overall data distribution using two principal components (PCs).
   - Goals include identifying similarities, outliers, understanding variable correlations, and assessing their contributions to each PC.

These interpretations provide valuable insights into how different variables contribute to each principal component and help in understanding the underlying structure of the data.

```
#biplot(PCAcountry, cex = 0.7)
biplot(PCAcountry, xlab = "1st Principal Component", ylab = "2nd Principal Component ", scale = F, cex = 0.5)
```




1st Principal Component: The plot generated from the biplot() function shows that there are no clear outliers in the dataset. However, it does reveal a strong correlation between the columns gdpp and income, as well as between the columns total_fer and child_mort. The biplot() function also allows us to visualize the contribution of each variable to each principal component, providing further insight into the relationship between the variables and PCs.

2nd Principal Component: The graphical representation shows that 'child_mort' and 'total_fer' are positively correlated and move in the same direction on PC1, while 'income', 'gdpp', and 'life_expec' are positively correlated and move in the opposite direction on PC1.

5.2 Choice for the number of PCA main vectors

As previously mentioned, the main objective of PCA is to perform dimensionality reduction. To identify the minimum number of principal components that can explain most of the variation in the data, the summary() function can be used.

The summary() function returns three key pieces of information:

-Standard deviation: the amount of variance captured by each principal component.
-Proportion of variance: the proportion of the total variance in the data captured by each principal component.
-Cumulative proportion: the cumulative proportion of the total variance explained by all the principal components, starting from PC1 to PC9.

```
summary(PCAcountry)
```

```
## Importance of components:
##                           PC1    PC2    PC3    PC4    PC5     PC6    PC7
## Standard deviation     2.0336 1.2435 1.0818 0.9974 0.8128 0.47284 0.3368
## Proportion of Variance 0.4595 0.1718 0.1300 0.1105 0.0734 0.02484 0.0126
## Cumulative Proportion  0.4595 0.6313 0.7614 0.8719 0.9453 0.97015 0.9828
##                           PC8     PC9
## Standard deviation     0.29718 0.25860
## Proportion of Variance 0.00981 0.00743
## Cumulative Proportion  0.99257 1.00000
```

That's correct. The number of principal components to be used in a PCA analysis is determined based on the amount of information needed to represent the data accurately. In this case, if a minimum of 80% of the information is required, then the number of principal components to be used would be from PC1-4. However, the exact number of principal components to be used may vary depending on the specific requirements of the project and the data being analyzed.

```
country_selected_pca <- data.frame(PCAcountry$x[,1:4])
head(country_selected_pca)
```

```
##                             PC1         PC2        PC3         PC4
## Afghanistan         -2.90428986 -0.09533386  0.7159652  1.00224038
## Albania              0.42862224  0.58639208  0.3324855 -1.15757715
## Algeria             -0.28436983  0.45380957 -1.2178421 -0.86551146
## Angola              -2.92362976 -1.69047094 -1.5204709  0.83710739
## Antigua and Barbuda  1.03047668 -0.13624894  0.2250441 -0.84452276
## Argentina            0.02234007  1.77385167 -0.8673884 -0.03685602
```

5.3 Apply on the given data

After selecting the appropriate number of principal components, they can be combined with the original data to create a new dataset that has fewer dimensions. This new dataset can then be used for further analysis, such as clustering or classification. The reduced number of dimensions can also make it easier to visualize and interpret the data.

```
PCAcountry <- country |>
  select_if(purrr::negate(is.numeric)) |>
  cbind(country_selected_pca)

glimpse(PCAcountry)
```

```
## Rows: 167
## Columns: 5
## $ country <fct> "Afghanistan", "Albania", "Algeria", "Angola", "Antigua and Ba~
## $ PC1     <dbl> -2.90428986, 0.42862224, -0.28436983, -2.92362976, 1.03047668,~
## $ PC2     <dbl> -0.09533386, 0.58639208, 0.45380957, -1.69047094, -0.13624894,~
## $ PC3     <dbl> 0.7159652, 0.3324855, -1.2178421, -1.5204709, 0.2250441, -0.86~
## $ PC4     <dbl> 1.00224038, -1.15757715, -0.86551146, 0.83710739, -0.84452276,~
```

```r
country_2 <- PCAcountry |> select(-country)

head(country_2)
```

```
##                            PC1         PC2        PC3         PC4
## Afghanistan          -2.90428986 -0.09533386  0.7159652  1.00224038
## Albania               0.42862224  0.58639208  0.3324855 -1.15757715
## Algeria              -0.28436983  0.45380957 -1.2178421 -0.86551146
## Angola               -2.92362976 -1.69047094 -1.5204709  0.83710739
## Antigua and Barbuda   1.03047668 -0.13624894  0.2250441 -0.84452276
## Argentina             0.02234007  1.77385167 -0.8673884 -0.03685602
```

6.Clustering_K-Means

6.1 Clustering by using PCA : Determining the optimum K-Value

One popular method for determining the optimum K-Value is the Elbow Method. This method involves visualizing the results of the clustering algorithm for different K-Values and identifying the K-Value where the total within sum of squares (WSS) starts to level off, creating a "bend" in the graph that looks like an elbow.

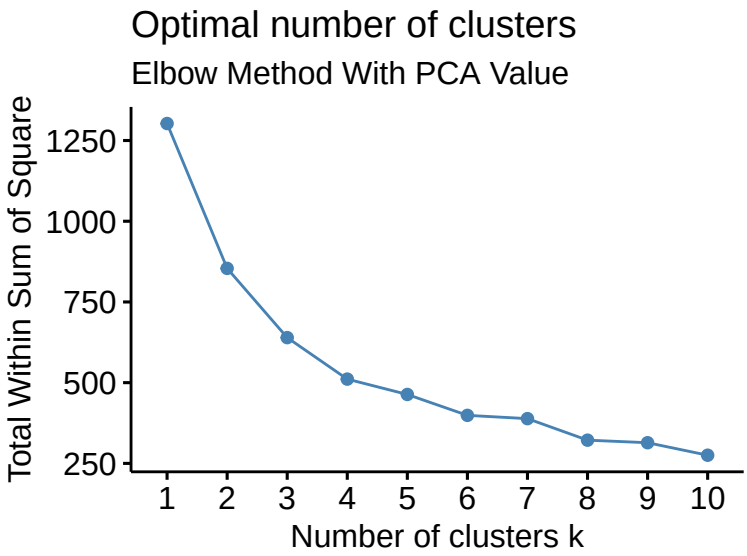
WSS is a measure of the variation that exists within each group, so a higher WSS value indicates a large degree of variability within the data set, while a lower value indicates that the data does not vary considerably from the mean value. The optimum K-Value is the point where increasing the number of K does not result in a considerable decrease in the total within sum of squares.

By using PCA to reduce the dimensionality of the data, we can first identify the principal components that explain the most variance in the data and then use these components to perform the clustering analysis. This approach can help to improve the accuracy of the clustering algorithm and reduce the computational complexity of the analysis.

a) Elbow Method

The Elbow Method is a popular approach to determine the optimal K-Value for clustering, which is the number of groups desired as the final result in K-Means. To apply this method, the fviz_nbclust() function is used to visualize the Within Sum of Square (WSS) values. WSS measures the variation that exists within each group, with higher values indicating greater variability in the data set and lower values indicating less variation from the mean value. The optimal K-Value is reached when increasing the number of K does not significantly decrease the total WSS.

```r
fviz_nbclust(country_2, kmeans, method = "wss") +
  labs(subtitle = "Elbow Method With PCA Value")
```



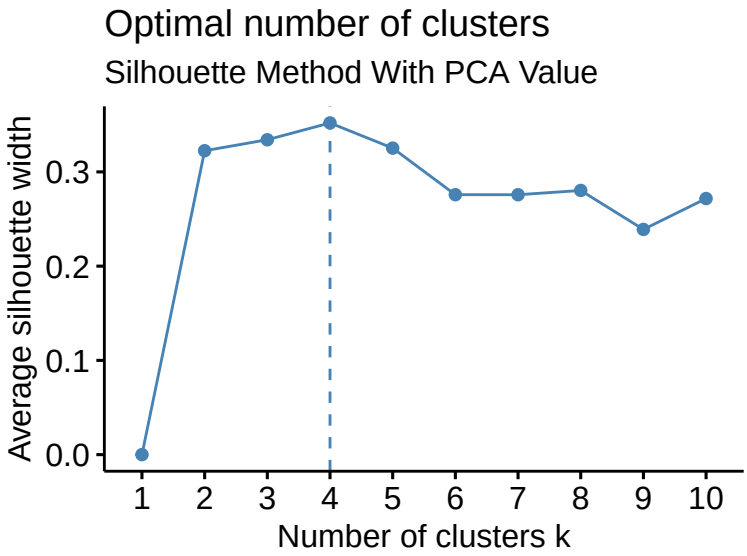

From the plots, it is evident that the optimum number of clusters or K value is 3. This is because, after k=3, increasing the number of clusters does not lead to a significant decrease in the total within sum of squares. Another approach to determine the optimum K value is to select the number of clusters at the "bend of an elbow" in the plot. However, this method can be considered biased as the location of the elbow is subjective and can vary from person to person.

b) Silhouette Method

The Silhouette Method is another approach to determine the optimal K-Value, which utilizes the same function as the Elbow Method. Function fviz_nbclust() will display a measure of the closeness of each point within a cluster to points in the neighboring clusters, allowing for visual assessment of parameters such as the number of clusters.

The optimal K-Value can be determined by identifying the highest Average Silhouette Width value or the peak value from the plot, indicating that the average distance between each cluster is not too close.

```
fviz_nbclust(country_2, kmeans, method = "silhouette") + labs(subtitle = "Silhouette Method With PCA Value")
```



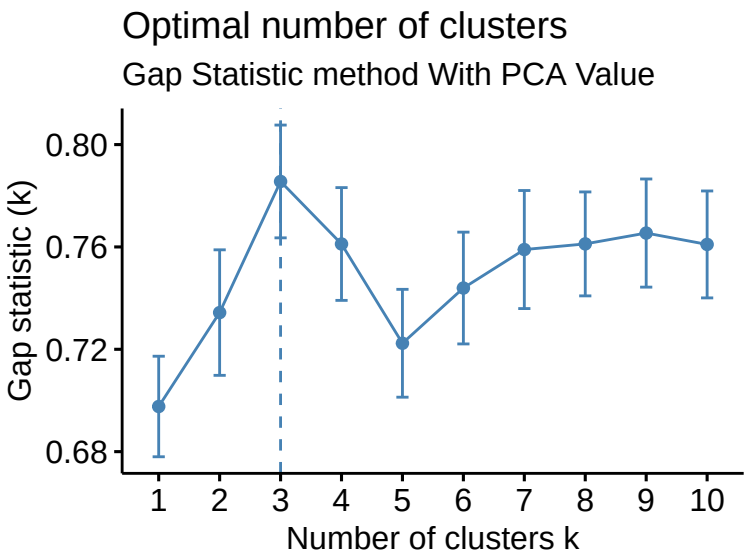

This statement correctly summarizes that based on the Silhouette Method, the optimal number of clusters for the given dataset is 4, as the Average Silhouette Width decreases after k=4.

c) Gap Statistic

The final technique is known as Gap Statistic. To observe the Gap Statistic, you can utilize the function fviz_nbclust(). Essentially, the Gap Statistic method selects the optimal value of K by determining the point at which the largest increase in within-cluster distance occurs. In other words, the optimal K-Value is the first instance at which the highest Gap Statistic value is achieved without dropping, as determined from the plot.

According to the results obtained from the Gap Statistic approach, the ideal value of K is 3.

```
fviz_nbclust(country_2, kmeans, method = "gap_stat") + labs(subtitle = "Gap Statistic method With PCA Value")
```




Out of the three methods described, two indicate that the optimal number of clusters is 3. Therefore, the data will be partitioned into 3 clusters.

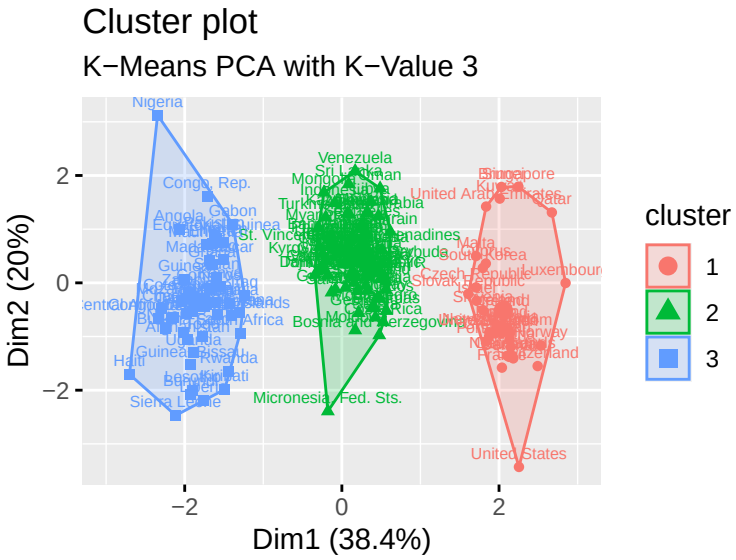
Note that: the aforementioned testing methods provide a means of determining the number of clusters, other factors such as business requirements or mutual agreement can also be taken into account.

Conducting clustering with PCA values.

```
set.seed(123)

kmpca <- kmeans(country_2, centers = 3)

country_2$cluster <- kmpca$cluster

unique(country_2$cluster)
```

```
## [1] 3 2 1
```

```
pcacluster <- fviz_cluster(kmpca, data = country_2, labelsize = 6) +
   labs(subtitle = "K-Means PCA with K-Value 3") + xlim(c(-3, 3))

pcacluster
```




RESULT from above : Hierarchical Cluster 1 = No help needed

Hierarchical Cluster 2 = Might need help

Hierarchical Cluster 3 = Help needed

For the detailed explanation, we can see the profiling results in the next chapter.

6.2 County Cluster Profiling The primary objective of profiling is to gain insights into the distinct features of each cluster. In this context, the aim is to determine which country cluster requires aid the most. To assess the characteristics of each cluster, the mean value of each column can be calculated.

```
country_2 |>group_by(cluster) |>
  summarise_all(mean)
```

```
## # A tibble: 3 x 5
##   cluster    PC1    PC2     PC3    PC4
##     <int>  <dbl>  <dbl>   <dbl>  <dbl>
## 1       1  2.79   0.234 -0.0424  0.895
## 2       2  0.204  0.147 -0.0506 -0.758
## 3       3 -2.39  -0.429  0.119   0.674
```

Here's an alternative phrasing for the sentence you provided:

It may be preferable to assign the clusters to their original values rather than using PCA for profiling, as this approach may be easier to interpret.

```r
country_observation <- scale # new object for clustering record
country_observation$cluster <- kmpca$cluster

country_observation |> group_by(cluster) |>
  summarise_all(mean) #K-means property
```

```
## # A tibble: 3 x 10
##   cluster child_mort exports health  imports income inflation life_expec
##     <int>      <dbl>   <dbl>  <dbl>    <dbl>  <dbl>     <dbl>      <dbl>
## 1       1     -0.828  0.632   0.765  0.191    1.49    -0.496       1.09
## 2       2     -0.411 -0.0228 -0.239  0.00919 -0.227   -0.0164      0.273
## 3       3      1.32  -0.421  -0.139 -0.155   -0.687    0.391      -1.27
## # i 2 more variables: total_fer <dbl>, gdpp <dbl>
```

Profiling Results:

Cluster 3: Economically, the population of countries in Cluster 3 exhibit poor performance, as evidenced by negative average values for income, GDP per capita, and inflation. Moreover, Cluster 3 countries are not doing well in the industrial sector, as average values for exports and imports are negative. Health-wise, the situation is particularly concerning, as Cluster 3 countries show a high average child mortality rate, as well as negative average values for health and life expectancy.

Cluster 2: Economically, countries in Cluster 2 are doing well, with positive average values for income, GDP per capita, and negative average value for inflation. Additionally, Cluster 2 can be considered a developed group of countries, given their high average values for exports and imports. Health-wise, the situation in Cluster 2 is mixed, with a high negative average for child mortality rate, but negative average values for health.

Cluster 1: Economically, the majority of countries in Cluster 1 have poor economic conditions, as indicated by negative average values for income and GDP per capita. Additionally, Cluster 1 countries face challenges in the industrial sector, with negative average values for exports, but positive averages for imports. Health-wise, Cluster 1 countries show a low average value for health, but a positive situation regarding child mortality rate and life expectancy.

## 7. Country Selection for Donations

### 7.1 Selection Based on Cluster 3's Overall Profile

According to the cluster profiling results, the countries in Cluster 3 are the most in need of aid compared to those in Clusters 2 and 1. To identify which countries within Cluster 3 should receive the donations first, we can compare their values to the average profiling value for Cluster 3. Countries with values lower than the average for Cluster 3 can be prioritized for aid, based on the mean values in right above table of the codes.

```r
country_observation %>%
  filter(child_mort > 1.322,
         exports < -0.42,
         health < -0.139,
         imports < -0.155,
         income < -0.687,
         inflation < 0.39,
         life_expec < -1.272,
         total_fer > 1.35,
         gdpp < -0.603)
```

```
##                           child_mort    exports     health    imports     income
## Cameroon                     1.72903 -0.6898063 -0.6136834 -0.8215842 -0.7513558
## Central African Republic     2.74567 -1.0692020 -1.0323467 -0.8422371 -0.8432738
##                            inflation life_expec total_fer       gdpp cluster
## Cameroon                  -0.5554817  -1.490547  1.428173 -0.6358417       3
## Central African Republic -0.5460216  -2.592516  1.494230 -0.6829809       3
```

### 7.2 Selection Based on Urgency of Economic and Health Parameters

We can also allocate donation funds based on the economic and health segments that are most in need of aid. For this purpose, we can examine columns that are suitable for assessment.

-Economic Sector For the economic sector, we can consider the following parameter columns: income, exports, imports, and GDP per capita (GDPP). To prioritize aid for countries in Cluster 1, the values for these columns should be lower than the average(the average table for each column for each cluster in the previous chapter) profiling value for Cluster 1.

```
#Parameter filter
ec <- country_observation |>
  filter(exports < -0.42,
         imports < -0.155,
         income < -0.687,
         gdpp < -0.603) |>
  select(income,exports,imports, gdpp)

ec
```

```
##                              income    exports    imports       gdpp
## Bangladesh                -0.7627678 -0.9159845 -1.0363751 -0.6659584
## Benin                     -0.7949287 -0.6314377 -0.4002635 -0.6659584
## Burkina Faso              -0.8151589 -0.7992473 -0.7141887 -0.6759428
## Burundi                   -0.8497059 -1.1742654 -0.3176516 -0.6947112
## Cameroon                  -0.7513558 -0.6898063 -0.8215842 -0.6358417
## Central African Republic -0.8432738 -1.0692020 -0.8422371 -0.6829809
## Eritrea                   -0.8156776 -1.3249293 -0.9744162 -0.6810168
## Gambia                    -0.8032283 -0.6314377 -0.1730808 -0.6766521
## Guinea-Bissau             -0.8172338 -0.9561129 -0.4828754 -0.6774704
## Kenya                     -0.7606929 -0.7445268 -0.5489649 -0.6545556
## Madagascar                -0.8172338 -0.5876612 -0.1606890 -0.6847814
## Malawi                    -0.8359079 -0.6679180 -0.4952672 -0.6822717
## Mali                      -0.7923350 -0.6679180 -0.4870060 -0.6686864
## Myanmar                   -0.6963711 -1.4956939 -1.9341227 -0.6534098
## Nepal                     -0.7861104 -1.1501884 -0.4333083 -0.6750153
## Rwanda                    -0.8193087 -1.0619059 -0.6976663 -0.6765975
## Senegal                   -0.7762546 -0.5913093 -0.2722151 -0.6527551
## Sierra Leone              -0.8260521 -0.8868002 -0.5117896 -0.6855452
## Sudan                     -0.7145264 -0.7810072 -1.2263824 -0.6265667
## Tanzania                  -0.7809231 -0.8174875 -0.7348417 -0.6690138
## Uganda                    -0.8094529 -0.8758561 -0.7554947 -0.6748516
```

It is difficult to determine which country is in most need of aid based solely on the economic sector results we have obtained so far. Although the 21 countries we have identified have an average value below that of the Cluster 3 profiling for columns such as income, exports, imports, and GDPP, it would be wise to prioritize countries with the lowest average value for these parameters.

To help identify which country should receive aid first, we can create visualizations of the data.

```
#transfrom negative value to positive to visualize
ec <- abs(ec)

#change index to columns to visualize
ec <- tibble::rownames_to_column(ec, "country")

head(ec)
```

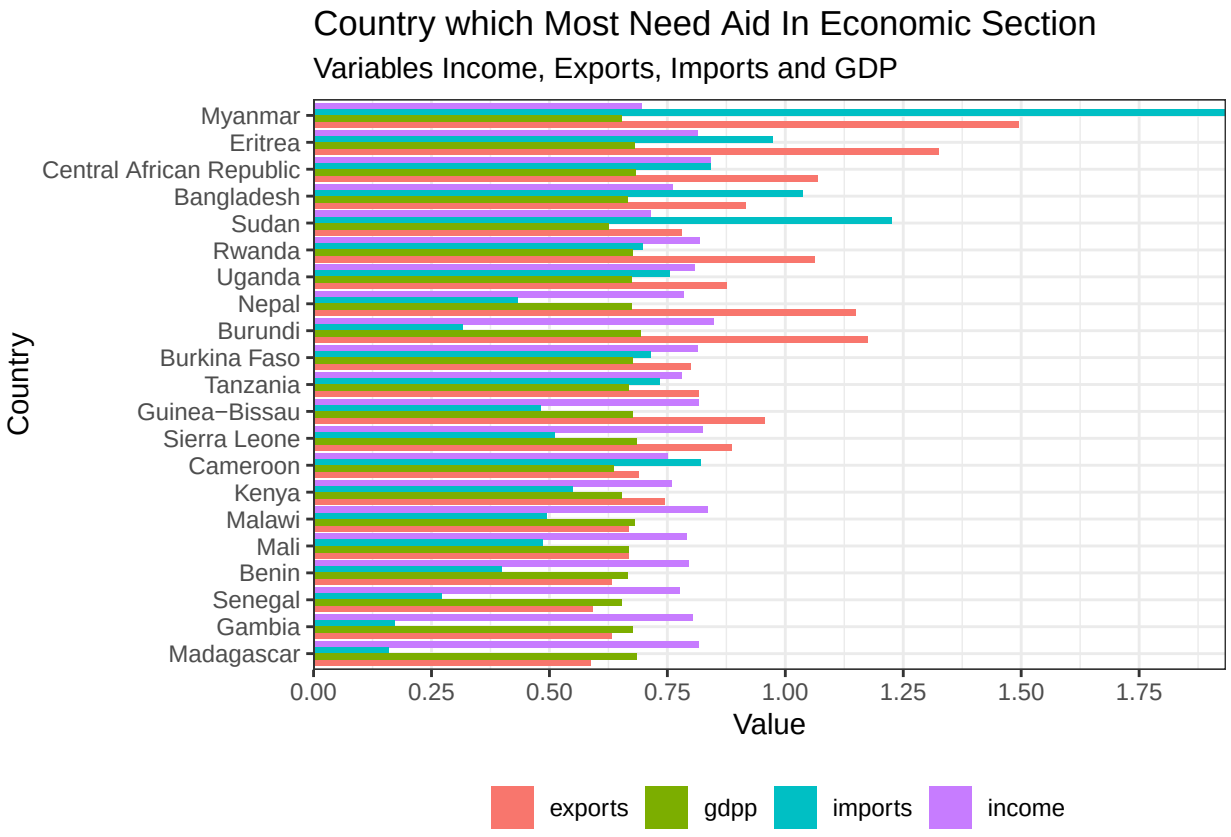
```
##                    country    income   exports   imports      gdpp
## 1               Bangladesh 0.7627678 0.9159845 1.0363751 0.6659584
## 2                    Benin 0.7949287 0.6314377 0.4002635 0.6659584
## 3             Burkina Faso 0.8151589 0.7992473 0.7141887 0.6759428
## 4                  Burundi 0.8497059 1.1742654 0.3176516 0.6947112
## 5                 Cameroon 0.7513558 0.6898063 0.8215842 0.6358417
## 6 Central African Republic 0.8432738 1.0692020 0.8422371 0.6829809
```

To make the visualization results easier to interpret, the function pivot_longer() can be implemented to combine the columns income, exports, imports, and gdpp.

```
economy_piv <- pivot_longer(data = ec, cols = c("income", "exports", "imports", "gdpp"))
head(economy_piv)
```

```
## # A tibble: 6 x 3
##   country    name    value
##   <chr>      <chr>   <dbl>
## 1 Bangladesh income  0.763
## 2 Bangladesh exports 0.916
## 3 Bangladesh imports 1.04
## 4 Bangladesh gdpp    0.666
## 5 Benin      income  0.795
## 6 Benin      exports 0.631
```

```
# Graph of the above 21 Counturies
ggplot(data = economy_piv, aes(x = value, y = reorder(country, value))) +
  scale_x_continuous(label = scales::comma,
                     expand = c(0,0),
                     breaks = seq(0, 2.5, 0.25)) +
  geom_col(aes(fill = name),position = "dodge") +
  labs(title = "Country which Most Need Aid In Economic Section",
       subtitle = "Variables Income, Exports, Imports and GDP",
       x = "Value",
       y = "Country",
       color = "") +
  theme_bw()  +
  theme(legend.position = "bottom",
        legend.title = element_blank())
```




-Health sector

In the health sector, the columns child_mort, health, and life_expec must have an average value lower than the average profiling value for cluster 3, based on the previous chapter column mean table values.

```r
#country_observation %>%
#  group_by(cluster) %>%
#  summarise_all(mean)

#Parameter filter
health <- country_observation %>%
  filter(child_mort > 1.322,
         health < -0.139,
         life_expec < -1.272) %>%
  select(child_mort, health, life_expec)

health
```

```
##                          child_mort     health life_expec
## Cameroon                   1.729030 -0.6136834  -1.490547
## Central African Republic   2.745670 -1.0323467  -2.592516
## Chad                       2.770466 -0.8321164  -1.580503
## Cote d'Ivoire              1.803418 -0.5517941  -1.602993
## Guinea                     1.753826 -0.6864944  -1.411835
## Mozambique                 1.555458 -0.5845590  -1.805395
## Niger                      2.100972 -0.6027618  -1.321878
```

Let's create a visualization to compare the child_mort, health, and life_expec columns and determine which countries are in the most urgent need of aid in the health sector among the above 7countries.

```r
#Change negative value into positive, for the sake of visualization
health <- abs(health)

#Change country section from index into columns, for the sake of visualization
health <- tibble::rownames_to_column(health, "country")

health
```

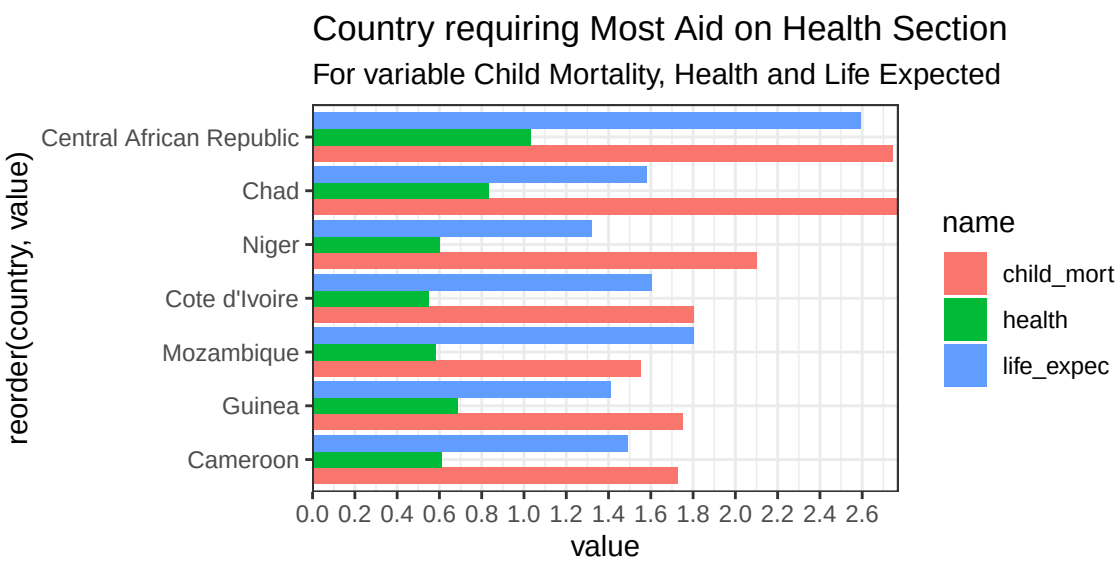
```
##                    country child_mort    health life_expec
## 1                Cameroon   1.729030 0.6136834   1.490547
## 2 Central African Republic   2.745670 1.0323467   2.592516
## 3                     Chad   2.770466 0.8321164   1.580503
## 4            Cote d'Ivoire   1.803418 0.5517941   1.602993
## 5                   Guinea   1.753826 0.6864944   1.411835
## 6               Mozambique   1.555458 0.5845590   1.805395
## 7                    Niger   2.100972 0.6027618   1.321878
```

To make the visualization results easier to interpret, we can use the pivot_longer() function to combine the child_mort, health, and life_expec columns.

```r
health_piv <- pivot_longer(data = health, cols = c("child_mort", "health", "life_expec"))
head(health_piv)
```

```
## # A tibble: 6 x 3
##   country                  name       value
##   <chr>                    <chr>      <dbl>
## 1 Cameroon                 child_mort 1.73
## 2 Cameroon                 health     0.614
## 3 Cameroon                 life_expec 1.49
## 4 Central African Republic child_mort 2.75
## 5 Central African Republic health     1.03
## 6 Central African Republic life_expec 2.59
```

```r
ggplot(data = health_piv, aes(x = value, y = reorder(country, value))) +
    scale_x_continuous(label = scales::comma, breaks = seq(0, 5, 0.2), expand = c(0,0)) +
  theme_bw()  +
  geom_col(aes(fill = name),position = "dodge") +
  labs(title = "Country requiring Most Aid on Health Section",
       subtitle = "For variable Child Mortality, Health and Life Expected")
```




8. Conclusion

To determine the ideal number of K-Values or groups for K-Means clustering, it is necessary to use various methods such as the Elbow Method, Silhouette Method, and Gap Statistic Method. It is important to use multiple methods as the results may vary, and in this case, two out of three methods suggest the ideal K-Value to be 3.

Once the ideal number of clusters is determined, the characteristics of each country in each cluster can be analyzed using Cluster Profiling. Based on the profiling results, countries in Cluster 1 are the most in need of aid when compared to countries in Cluster 2 and Cluster 3

Cluster 1 profiling results reveal that countries in this cluster are in dire need of aid, particularly in the economic and health sectors. Economic growth in these countries, as indicated by the negative results in the exports, imports, and gdpp columns, is poor or even stagnant. In terms of health, the mortality rate among children under 5 years old is alarmingly high, while the life expectancy figures suggest that a significant portion of the population is not expected to live long. These poor health figures may be contributing to the overall dismal economic situation in these countries.

There are two approaches to selecting countries in Cluster 3 for aid assistance:

1) Allocating all funds to countries with an average below the profiling cluster 3. Based on the filtering results above, it is evident that there are two countries, Cameroon and Central African Republic, that have an average below the profiling cluster 3.

2) Determine which countries to assist based on Economic & Health Urgency Parameters. In order to determine which countries to assist based on economic and health urgency parameters, there are two methods that can be applied. The first method involves allocating all funds to countries that have an average below the profiling cluster 3. However, this method only identifies two countries, Cameroon and Central African Republic, that can be assisted. The second method involves considering countries that have an average below the average profiling cluster 3 for either the economic or health segment, resulting in a total of 21 countries needing economic assistance and 7 countries needing health assistance. Prioritizing which countries to assist can be based on their ranking in each segment. For the economic segment, countries like Eritrea, Central African Republic, and Sudan can be assisted first, while for the health segment, countries like Central African Republic, Chad, and Niger can be prioritized.

In conclusion, the second method is a more effective approach as it enables more countries to receive aid, and aid can be given more accurately based on the specific segment in which a country needs assistance and how urgently they require it. The countries that rank within the top three in each segment can be prioritized to receive aid first.

The RESULT RANKING of country will received the earliest of aid in each sector:

Ranking Economic sector: 1st=Myanmar, 2nd=Sudan, 3rd=Bangladesh Ranking Health sector: 1st=Chad, 2nd=Central African Republic, 3rd=Niger

## 9. Limitation and future work

We need more reasonable and sophisticated criteria to filter the country from the table of K-mean clustering I mentioned, to set priority for the people who struggling with their survival daily. In economic sector, I put more weight to the income variable to set ranking from the filterking result of K-mean clustering. Also in Health sector, I put more weight to the child-mortality. Though Cameroon, Central African Republic are picked to be aid-need most countries from all section filtering, specific section filtering resulted in different country ranking as the previous chapter, which is different from these two countries. There are too many column variables, so we should analyze the importance between the variables to set ranking.

I want to simulate many of countries with bootstrap method(to have effect as using big data), and selecting appropriate regression model with tree-based method for using various statistics methods to train data. By these method, I can find more fitted model and groups to select proper countries for aid.