CLUSTERING COUNTRIES BY UNSUPERVISED LEARNING FOR HELP INTERNATIONAL

NAME - PAULAMI SANYAL

AUGUST 2022

IBM MACHINE LEARNING PROFESSIONAL CERTIFICATE

UNSUPERVISED MACHINE LEARNING – FINAL ASSIGNMENT

OBJECTIVE

- The objective is to categorize countries using socio-economic and health factors that determine the overall development of the country

- Problem Statement: HELP International have been able to raise around $ 10 million. Now the CEO of the NGO needs to decide how to use this money strategically and effectively. So, CEO has to make decision to choose the countries that are in the direst need of aid. Hence, your Job as a Data scientist is to categorize the countries using some socio-economic and health factors that determine the overall development of the country. Then you need to suggest the countries which the CEO needs to focus on the most.

DATASET DESCRIPTION

Our dataset contains 167 rows and 10 columns, i.e., 167 data points and 10 features. Following are the features:

1. country – The name of the Country
2. child_mort - Death of children under 5 years of age per 1000 live births
3. exports - Exports of goods and services per capita. Given as %age of the GDP per capita
4. health - Total health spending per capita. Given as %age of GDP per capita
5. imports - Imports of goods and services per capita. Given as %age of the GDP per capita
6. income - Net income per person
7. inflation - The measurement of the annual growth rate of the Total GDP
8. life_expec - The average number of years a new born child would live if the current mortality patterns are to remain the same
9. total_fer - The number of children that would be born to each woman if the current age-fertility rates remain the same.
10. gdpp - The GDP per capita. Calculated as the Total GDP divided by the total population.

DATA EXPLORATION

- Our dataset had the column country which couldn't add to the model. Hence, we removed it.

```
1 data.head()
```

	country	child_mort	exports	health	imports	income	inflation	life_expec	total_fer	gdpp
0	Afghanistan	90.2	10.0	7.58	44.9	1610	9.44	56.2	5.82	553
1	Albania	16.6	28.0	6.55	48.6	9930	4.49	76.3	1.65	4090
2	Algeria	27.3	38.4	4.17	31.4	12900	16.10	76.5	2.89	4460
3	Angola	119.0	62.3	2.85	42.9	5900	22.40	60.1	6.16	3530
4	Antigua and Barbuda	10.3	45.5	6.03	58.9	19100	1.44	76.8	2.13	12200

- All other features were of type float64, except income, so we converted it to type float64 as well.

- There were no null values in any of the features.

```
1 data.isnull().any()
```

```
child_mort     False
exports        False
health         False
imports        False
income         False
inflation      False
life_expec     False
total_fer      False
gdpp           False
dtype: bool
```

DATA EXPLORATION

- There were huge gaps in values for various features. Hence, we scaled the dataset using StandardScaler.

```
1 data.describe()
```

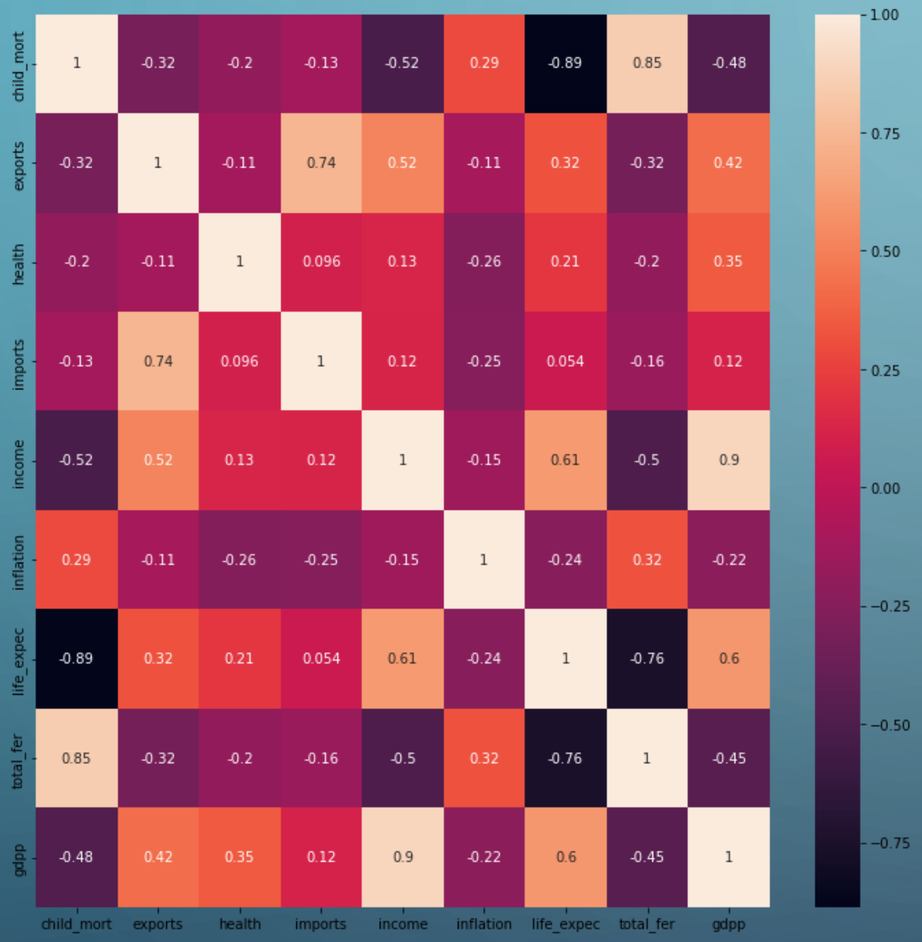
	child_mort	exports	health	imports	income	inflation	life_expec	total_fer	gdpp
count	167.000000	167.000000	167.000000	167.000000	167.000000	167.000000	167.000000	167.000000	167.000000
mean	38.270060	41.108976	6.815689	46.890215	17144.688623	7.781832	70.555689	2.947964	12964.155689
std	40.328931	27.412010	2.746837	24.209589	19278.067698	10.570704	8.893172	1.513848	18328.704809
min	2.600000	0.109000	1.810000	0.065900	609.000000	-4.210000	32.100000	1.150000	231.000000
25%	8.250000	23.800000	4.920000	30.200000	3355.000000	1.810000	65.300000	1.795000	1330.000000
50%	19.300000	35.000000	6.320000	43.300000	9960.000000	5.390000	73.100000	2.410000	4660.000000
75%	62.100000	51.350000	8.600000	58.750000	22800.000000	10.750000	76.800000	3.880000	14050.000000
max	208.000000	200.000000	17.900000	174.000000	125000.000000	104.000000	82.800000	7.490000	105000.000000

```
1 data.skew()

child_mort    1.450774
exports       2.445824
health        0.705746
imports       1.905276
income        2.231480
inflation     5.154049
life_expec   -0.970996
total_fer     0.967092
gdpp          2.218051
dtype: float64
```

- We see positive skewness with all the features, except life_expec initially. However, with log transformation, that turned into negative skewness of much lower degree, except for health and total_fer features, which were still positively skewed. However, in order to keep the values genuine, all the original skewness was kept intact.

DATA EXPLORATION



- Imports and Exports have very high positive correlation. (+0.74)
- Life Expectancy and Child mortality has very high negative correlation (-0.89)
- Total Fertility and child mortality has a high correlation. (+0.85)
- GDPP and Income has the highest positive correlation (+0.9) – as GDP is directly proportional to the income levels of people
- Life Expectancy has fairly high correlation with Income (+0.61) – this would mean as living standards improve, so does life expectancy
- GDPP has high correlation with Life Expectancy (+0.6)
- Total Fertility has fairly high negative correlation with Life Expectancy
- (-0.76). This would mean life expectancy of children of a woman having more children would reduce.

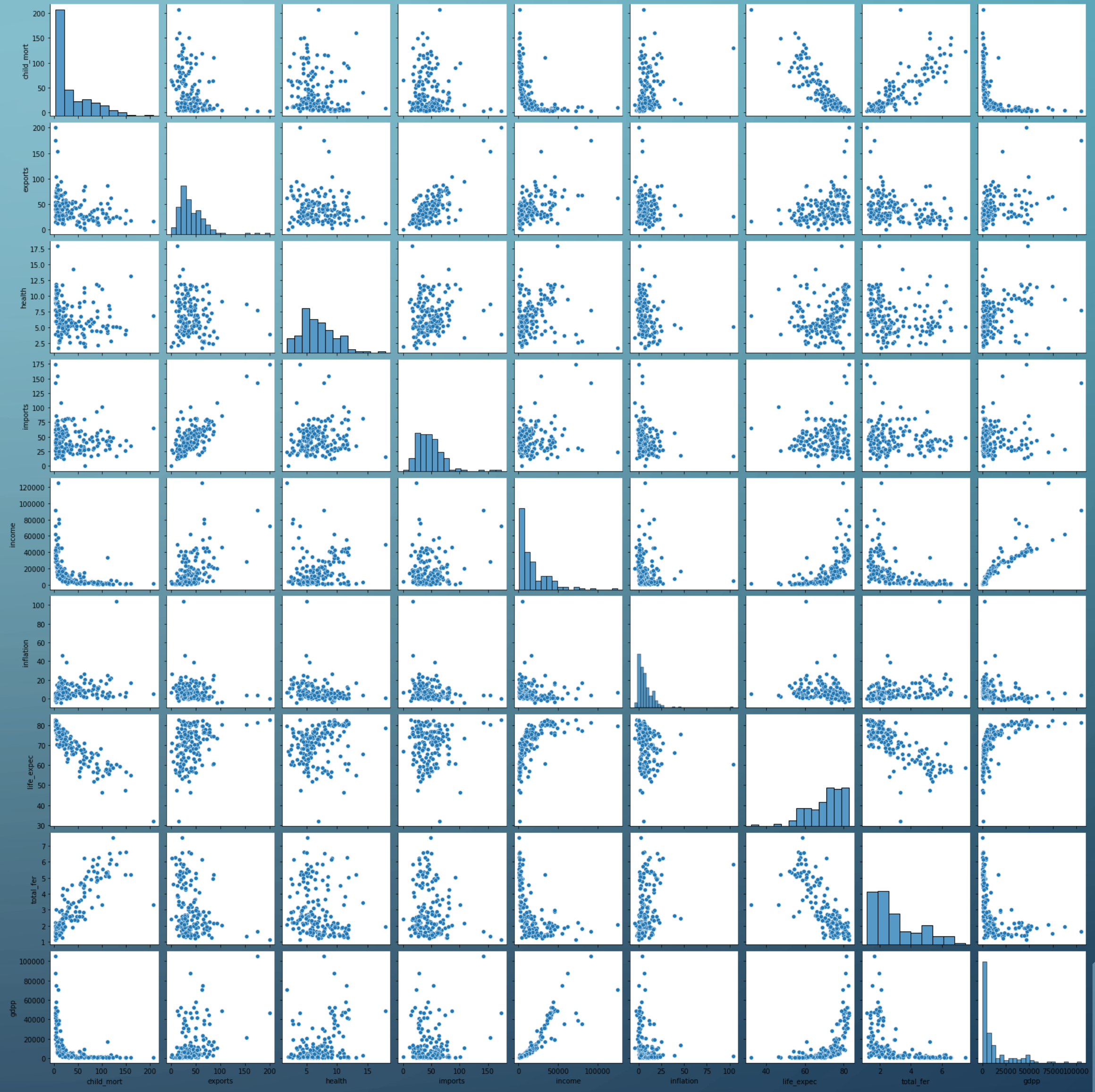
DATA EXPLORATION

Histograms:

- Most of the data is right skewed except for life expectancy which is left skewed.
- There are two peaks in GDPP and total fertility suggesting at least 2 clusters can be formed in the data.
- All of the plots suggest there are outliers.

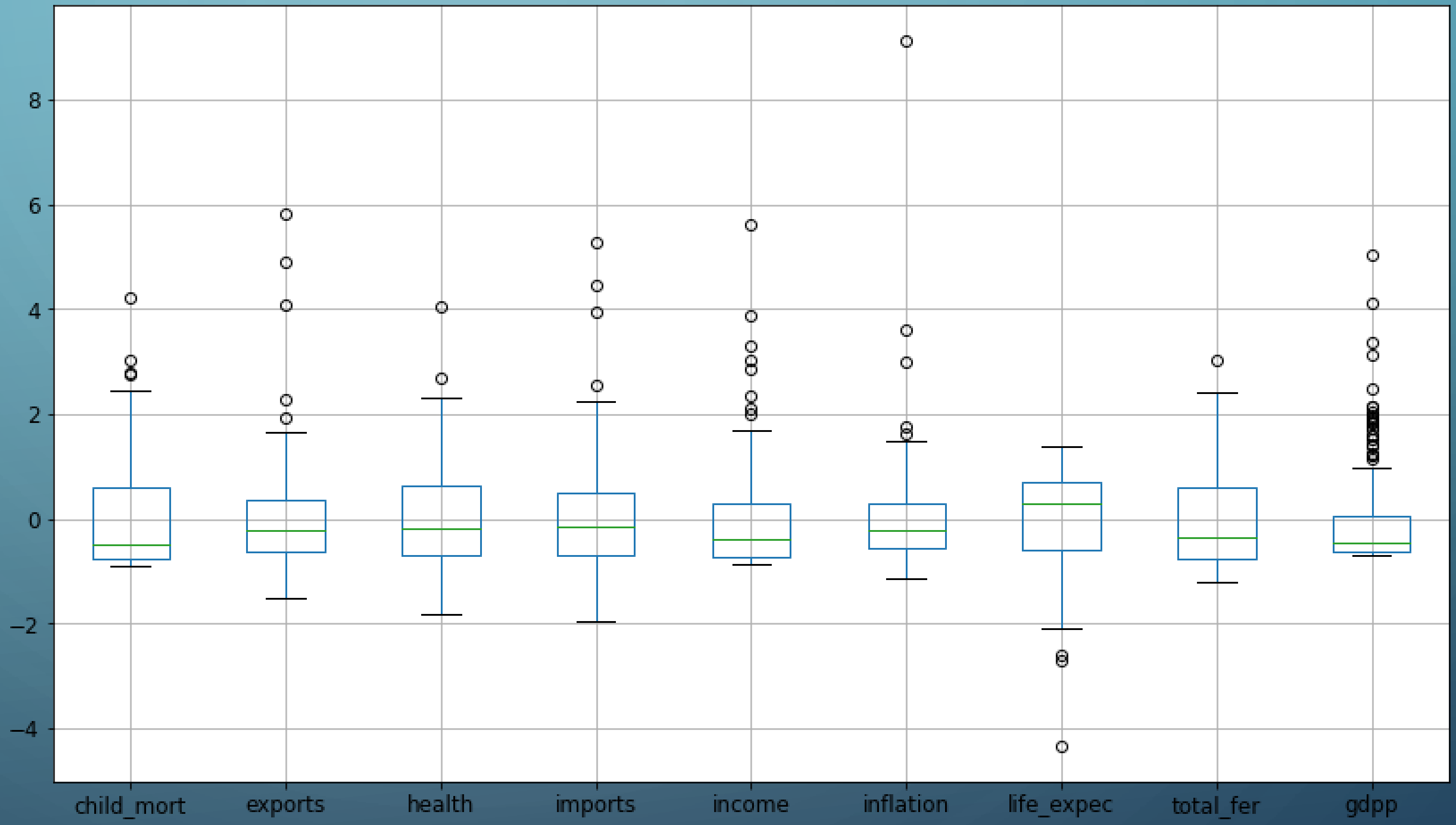
Scatter Plot:

- Income and GDPP have high correlation.
- All Countries with higher GDPP have low child mortality, total fertility and high life expectancy and lower inflation.
- Also, all countries with higher income have lower child mortality

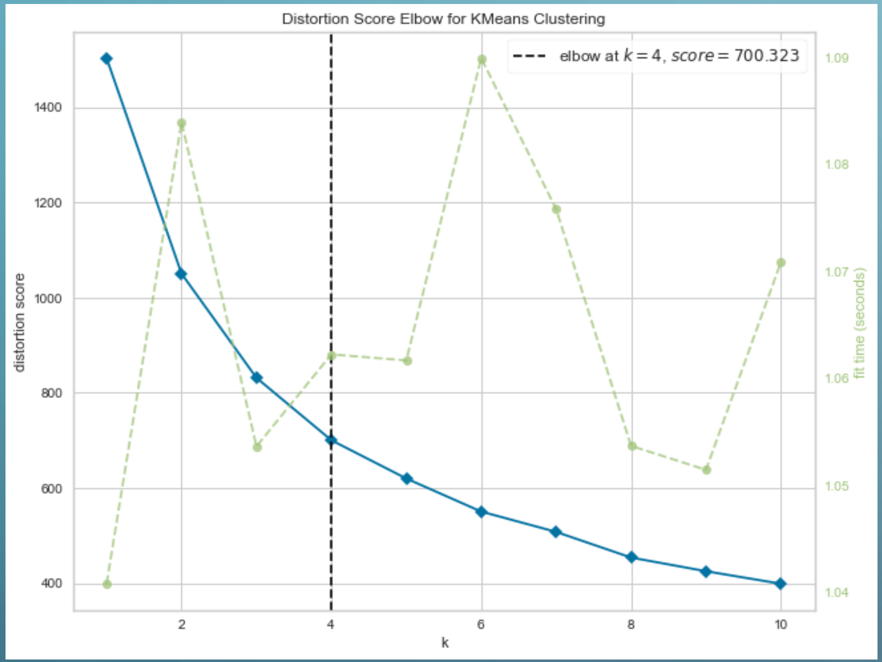


DATA EXPLORATION

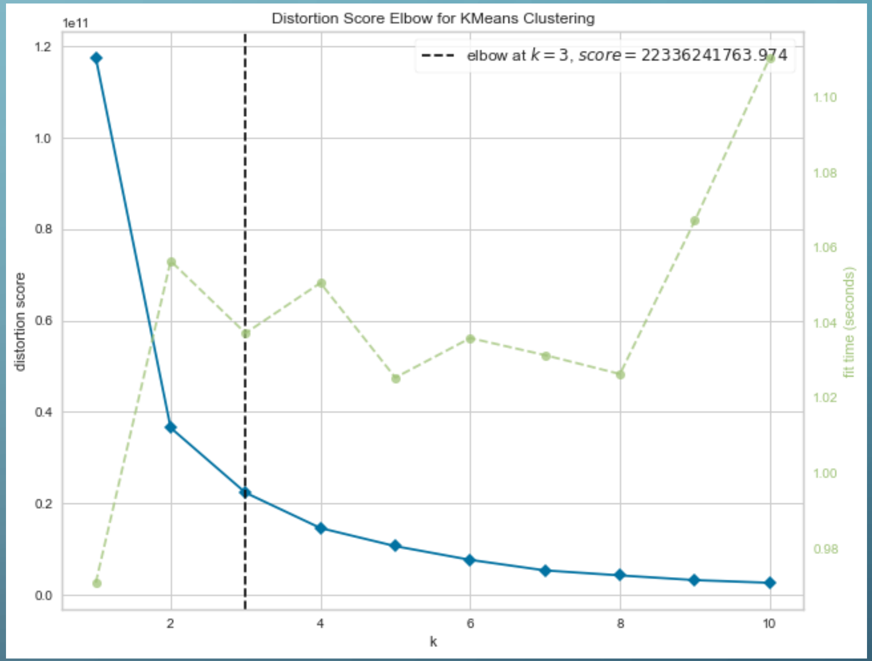
- Except for Life Expectancy, all the boxplot have outliers only on the upper end. (This concurs with the observations made in pairplot)
- GDPP has a lot of positive outliers.
- Inflation has few outliers but, one has very high value which will affect the distribution.
- Outliers won't be tampered with as they may contain genuine insight about the countries



MODELLING: KMEANS

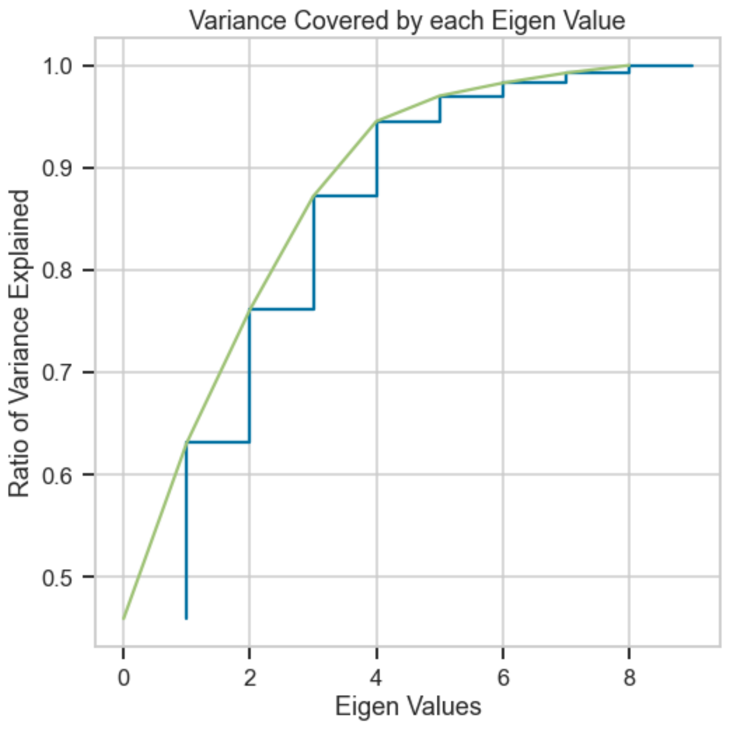


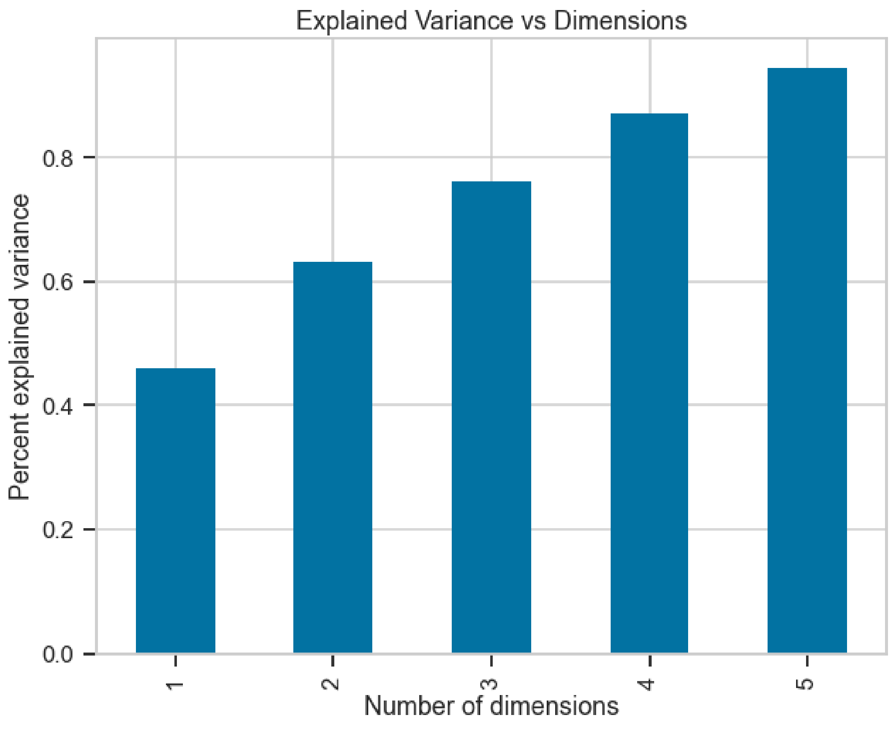
For scaled data we found 4 clusters to be optimal



For unscaled data, we found 3 clusters to be optimal

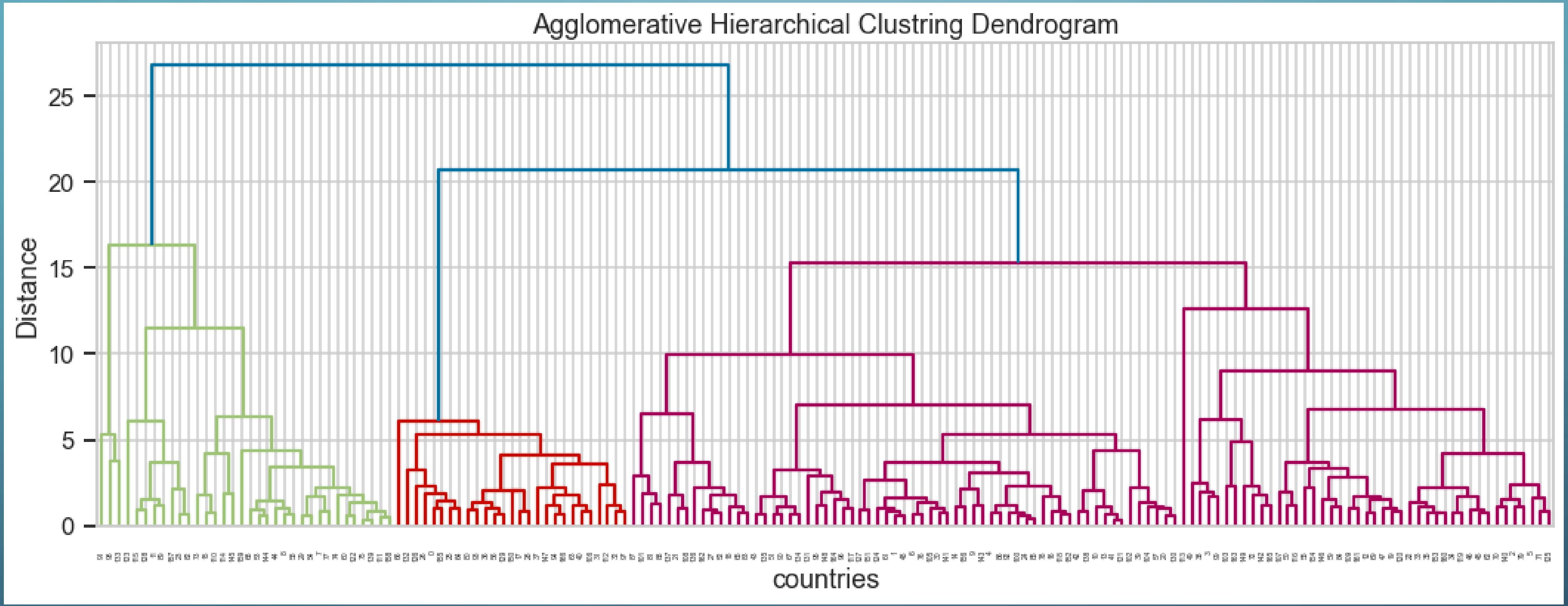
MODELLING: PCA





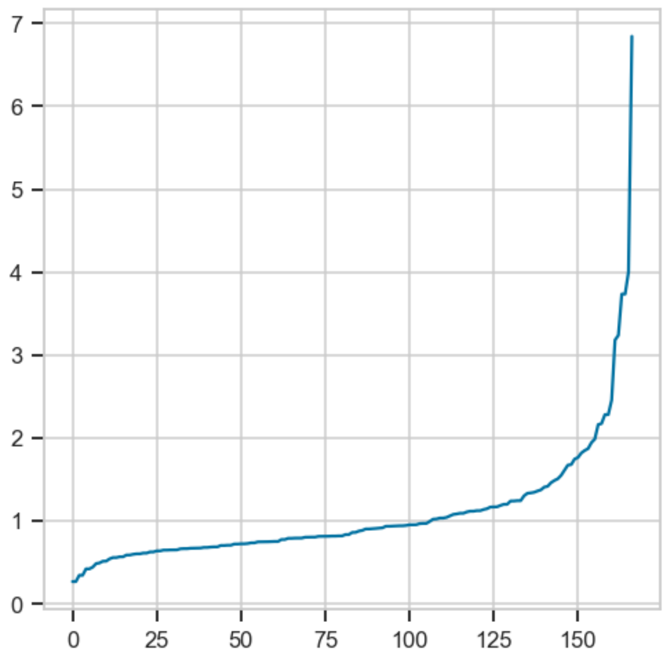
- Variance covered tapers off after the 4th Eigen Value. It covers more than 90% of the variance. So, even with PCA 4 clusters is found to be optimal number of clusters.

MODELLING: HIERARCHICAL CLUSTERING



- Hierarchical Agglomerative Clustering gave us 3 as the optimal number of clusters

MODELLING: DBSCAN



- Optimal value of Epsilon is 1.3 as it forms elbow like shape around that point.

- This gives 3 clusters of 0, 1, and 2. Countries with -1 values are noisy points and are not part of any clusters.

```python
from sklearn.cluster import DBSCAN

db = DBSCAN(eps = 1.3, min_samples = 8)# minimum samples is set to 8
db.fit(sData)
sPredDB = pd.Series(db.labels_)
print(pd.concat({'count' : sPredDB.value_counts(),
                 'percent' : round(sPredDB.value_counts(normalize = True)*100, 2)},
                axis = 1 ))
```

	count	percent
0	76	45.51
-1	53	31.74
2	20	11.98
1	18	10.78

RECOMMENDATION

Agglomerative Clustering:

```
print('Silhouette Score:', '%.2f'%sil_score(sData, sPredAGC))
print('Davies Bouldin Score:', '%.2f'%davies_bouldin_score(sData, sPredAGC))
```

```
Silhouette Score: 0.25
Davies Bouldin Score: 1.30
```

KMeans Clustering:

```
from sklearn.metrics import davies_bouldin_score

print('Silhouette Score:', '%.2f'%sil_score(sData, sPredKM))
print('Davies Bouldin Score:', '%.2f'%davies_bouldin_score(sData, sPredKM))
```

```
Silhouette Score: 0.30
Davies Bouldin Score: 1.05
```

DBSCAN:

```
print('Silhouette Score:', '%.2f'%sil_score(sData, sPredDB))
print('Davies Bouldin Score:', '%.2f'%davies_bouldin_score(sData, sPredDB))
```

```
Silhouette Score: 0.13
Davies Bouldin Score: 2.24
```

Selecting the best Clustering Method:

- Using Silhouette Score : Higher values are better. Values range from -1 to 1.
- Using Davies Bouldin Score : The minimum score is zero, with lower values indicating better clustering.

- DBSCAN has the lowest Silhouette score and a very high Davies Bouldin score which indicates overall clustering is not optimal.

- Also, DBSCAN put a lot of countries(53) in noisy group and we cannot have any country that needs help be ignored. So we wont be using clusters formed by DBSCAN

- KMeans Clustering gave the best Silhouette score and Davies Bouldin score of 0.3 and 1.04 respectively. Hence, is the optimal algorithm.

SUMMARY

```
print(dataKM.Class.value_counts())
dataKM.groupby('Class').mean()
```

```
1    85
2    47
3    32
0     3
Name: Class, dtype: int64
```

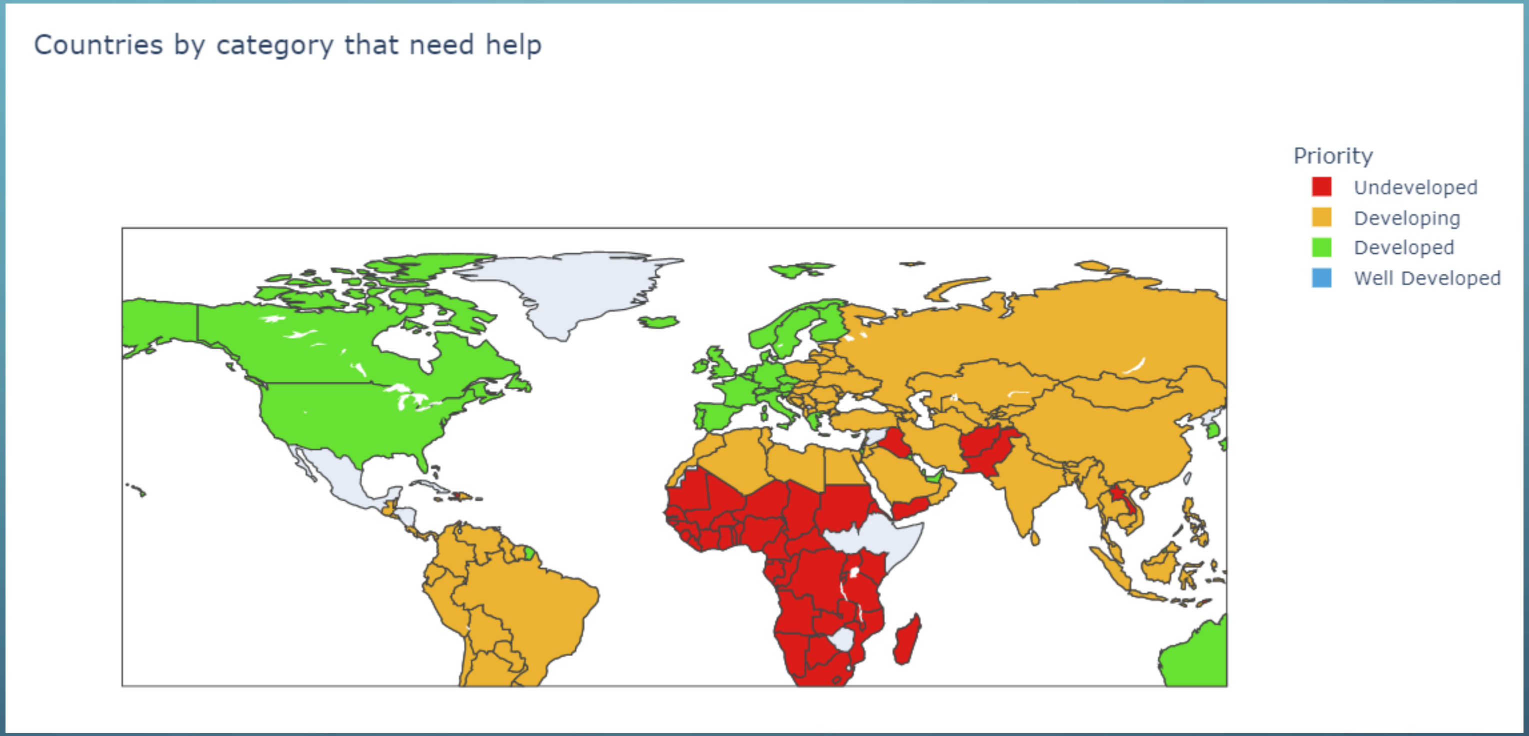
Class	child_mort	exports	health	imports	income	inflation	life_expec	total_fer	gdpp
0	4.133333	176.000000	6.793333	156.666667	64033.333333	2.468000	81.433333	1.380000	57566.666667
1	21.690588	41.073988	6.197059	47.914893	12671.411765	7.609341	72.871765	2.300706	6519.552941
2	92.961702	29.151277	6.388511	42.323404	3942.404255	12.019681	59.187234	5.008085	1922.382979
3	5.181250	46.118750	9.088437	40.584375	44021.875000	2.513844	80.081250	1.788437	42118.750000

Development	Classification	Priority
Underdeveloped	Need Help	1
Developing	Might Need Help	2
Developed	Do not need immediate help	3
Well Developed	Do not need help	4

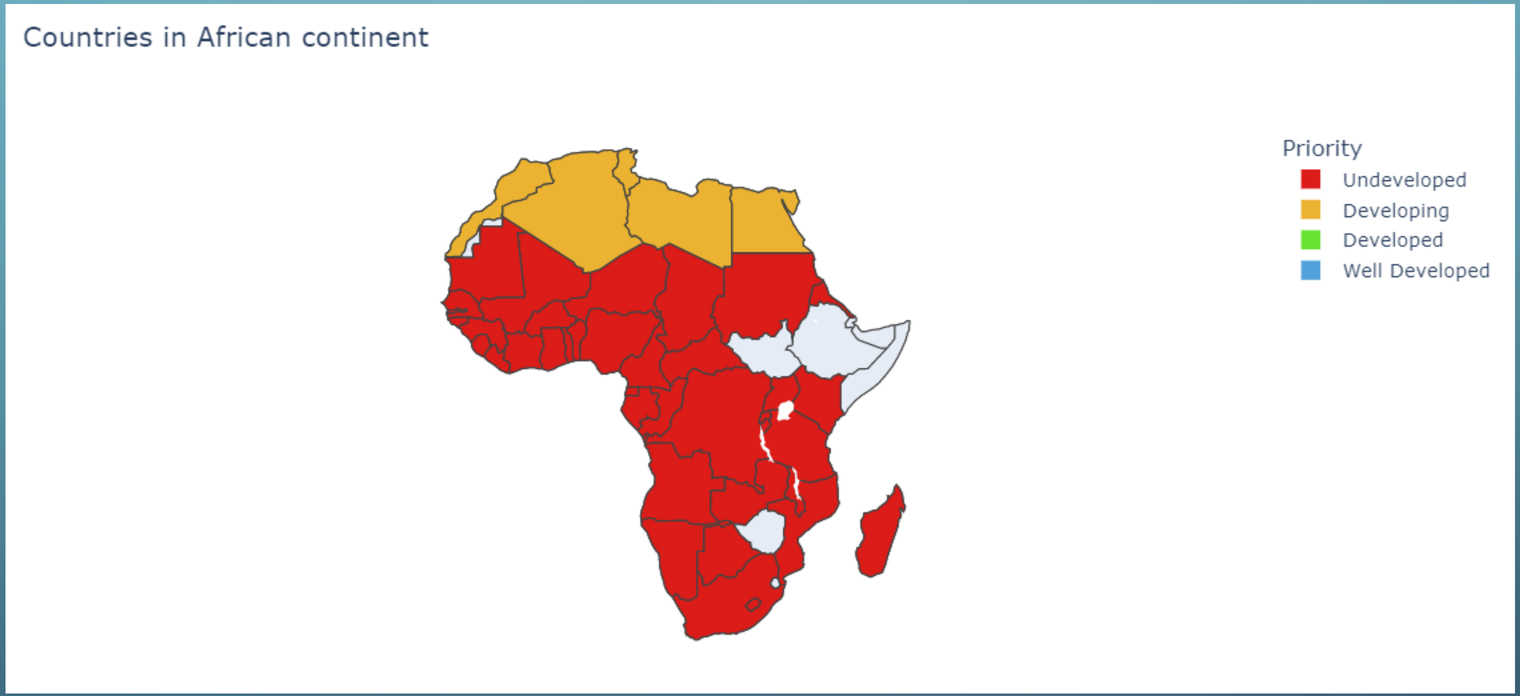
Findings:

- Class 2 with 47 countries has highest child mortality rate, lowest GDPP & Income, and its inflation is significantly higher than other groups. Countries in this group will be most disadvantaged and Undeveloped. The need help the most and should be 1st priority.
- Class 0 with 3 countries: It has the lowest child mortality rate, highest GDPP & Income, and has the lowest Inflation. This group contains most Well Developed countries with stable economies and health-care given that it has the highest life expectancy. These countries do not need any help and should have least priority in the list of countries requiring aid.
- Class 1 with 85 countries has the 2nd highest child mortality rate, 2nd lowest GDPP & Income, and even though its inflation is 2nd highest, its not significantly high. These countries are developing countries. These countries might need help and should be 2nd priority in the list of countries requiring aid.
- Class 3 with 32 countries has 2nd lowest child mortality rate, 2nd highest GDPP & Income. Also its inflation is 2nd lowest. It has significantly higher spendings on health. This group has Developed countries. These countries do not need help and can be 3rd priority in the list of countries requiring aid.

SUMMARY

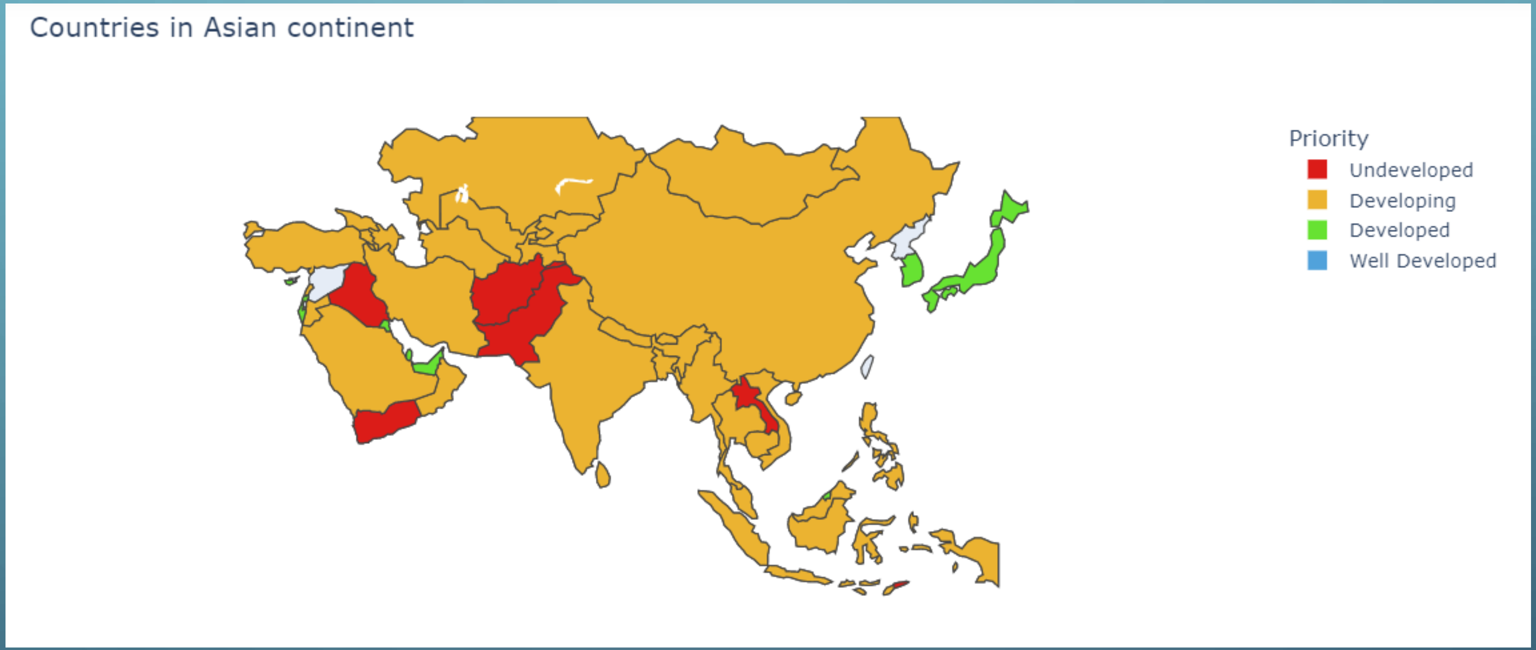

Countries by category that need help
Priority
Undeveloped
Developing
Developed
Well Developed

SUMMARY



Countries in African continent
Priority
Undeveloped
Developing
Developed
Well Developed

SUMMARY


Countries in Asian continent
Priority
Undeveloped
Developing
Developed
Well Developed

SUGGESTIONS

- GridSearchCV could have been used to further look into various hyperparameters in the different algorithms.
- Further other algorithms could have been looked into.
- Various others features like living standards in different geographical locations like rural, semi-urban, etc., could also have been used while building the model.

THANK YOU!

Reference: kaggle.com