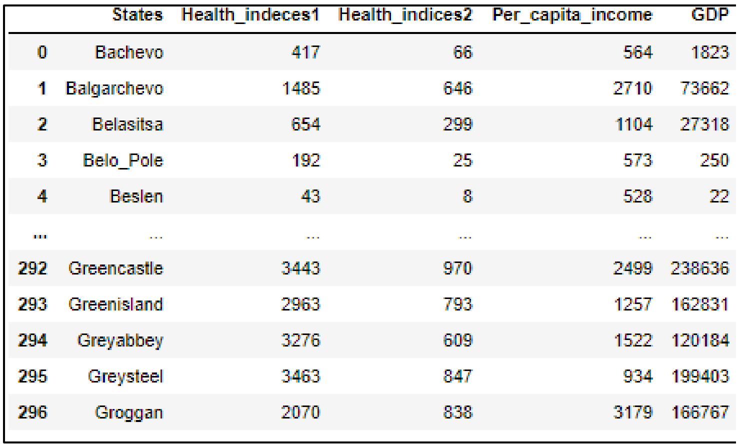
Clustering – Hierarchical and K Means

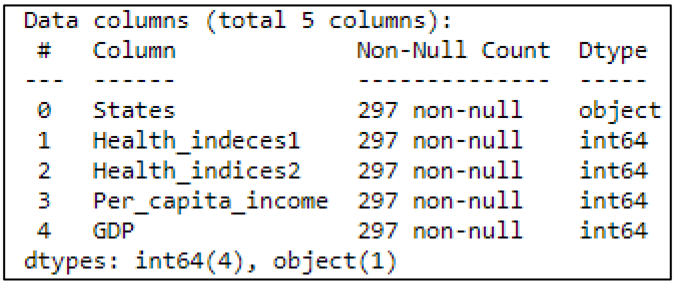
The dataset given is about the Health and economic conditions in different States of a country. The Group States based on how similar their situation is, so as to provide these groups to the government so that appropriate measures can be taken to escalate their Health and Economic conditions.

1. Read the data and do exploratory data analysis.



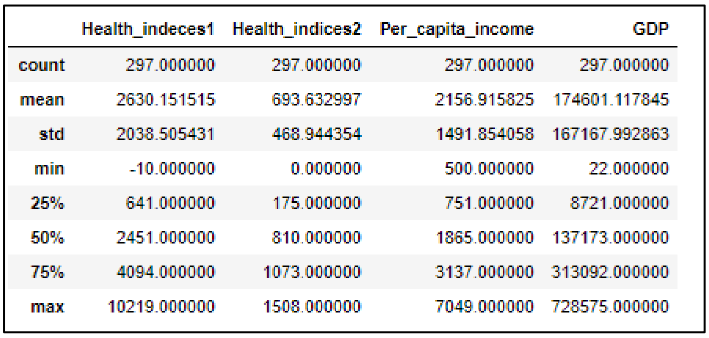
	States	Health_indeces1	Health_indices2	Per_capita_income	GDP
0	Bachevo	417	66	564	1823
1	Balgarchevo	1485	646	2710	73662
2	Belasitsa	654	299	1104	27318
3	Belo_Pole	192	25	573	250
4	Beslen	43	8	528	22
...	...	...	...	...	...
292	Greencastle	3443	970	2499	238636
293	Greenisland	2963	793	1257	162831
294	Greyabbey	3276	609	1522	120184
295	Greysteel	3463	847	934	199403
296	Groggan	2070	838	3179	166767

The dataset has 297 records, and 5 features. Health_indices1, Health_indices2, Per _capita_income and GDP are integers, while the data type of 'States' is object. States is categorical feature with 296 unique values, but we will be **dropping the 'States'** feature as in case of clustering, we have to calculate the distances between data points, and numerical data would be preferred.



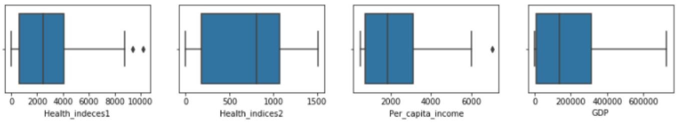
```
Data columns (total 5 columns):
 #   Column             Non-Null Count  Dtype
---  ------             --------------  -----
 0   States             297 non-null    object
 1   Health_indeces1    297 non-null    int64
 2   Health_indices2    297 non-null    int64
 3   Per_capita_income  297 non-null    int64
 4   GDP                297 non-null    int64
dtypes: int64(4), object(1)
```

Exploring the dataset:



	Health_indeces1	Health_indices2	Per_capita_income	GDP
count	297.000000	297.000000	297.000000	297.000000
mean	2630.151515	693.632997	2156.915825	174601.117845
std	2038.505431	468.944354	1491.854058	167167.992863
min	-10.000000	0.000000	500.000000	22.000000
25%	641.000000	175.000000	751.000000	8721.000000
50%	2451.000000	810.000000	1865.000000	137173.000000
75%	4094.000000	1073.000000	3137.000000	313092.000000
max	10219.000000	1508.000000	7049.000000	728575.000000

As we can see from above data, all 4 features have varying scales, and Health_indices1, and GDP having extremely high standard deviation, almost equal to the mean. We will tackle this problem shortly.



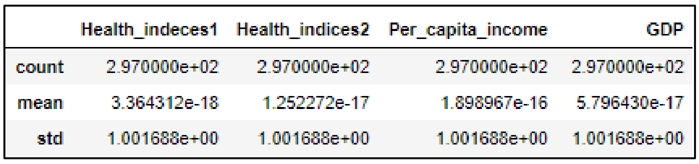
Health_Indeces1 and Per_capita_income have 2 and 1 outlier each, which is not good for clustering, so we will be replacing the outliers with their respective median values.

2. **Scaling of data**

Yes scaling is required in case of clustering, as we calculate the distance between the data points, and cluster on the basis of the distance. Here we have mixed numerical data with varied values and varied scales, therefore values won't be comparable. We will go for z- standardization, where we will be centering the features at mean = 0 and standard deviation as 1.

$$Z = \frac{x - \mu}{\sigma}$$

Z = standard score
x = observed value
μ = mean of the sample
σ = standard deviation of the sample

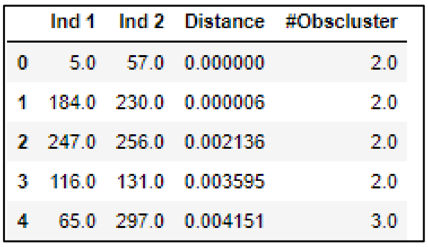


	Health_indeces1	Health_indices2	Per_capita_income	GDP
count	2.970000e+02	2.970000e+02	2.970000e+02	2.970000e+02
mean	3.364312e-18	1.252272e-17	1.898967e-16	5.796430e-17
std	1.001688e+00	1.001688e+00	1.001688e+00	1.001688e+00

Standardized data, with mean = 0 at standard deviation = 1

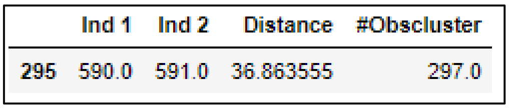
3. Apply hierarchical clustering to scaled data.

In Agglomerative hierarchical clustering technique, each point is initially considered a single cluster and merge with other clusters in each iteration to eventually form 1 cluster. To calculate the proximity, we will consider 'ward' linkage method. Ward linkage calculates the root mean square distance to measure the similarity between two clusters. Metric used will be Euclidean.



	Ind 1	Ind 2	Distance	#Obscluster
0	5.0	57.0	0.000000	2.0
1	184.0	230.0	0.000006	2.0
2	247.0	256.0	0.002136	2.0
3	116.0	131.0	0.003595	2.0
4	65.0	297.0	0.004151	3.0

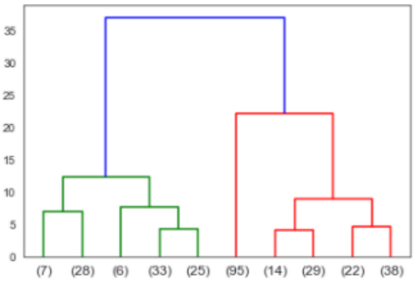
The first row is the first iteration, where observations from Ind 1 and Ind 2 are combined to form a cluster, Distance gives the Euclidean distance between the two points, and #Obscluster gives the number of observations at that time in this cluster.



	Ind 1	Ind 2	Distance	#Obscluster
295	590.0	591.0	36.863555	297.0

This is the last row, where after n-1 iterations, 1 cluster is formed, which contains all n observations. Next we will create a dendrogram, which is a visual tree like representation of the clustered data.

Dendrogram:



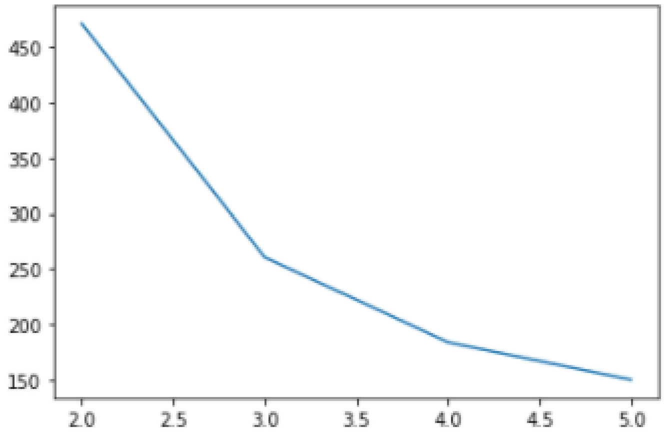
X axis shows the clusters and number of observations in those clusters, y axis shows the distance. Each data point is initially considered as a cluster. Based on similarity of clusters(linkage conditions defined) 2 points merge at a vertical distance to create a new cluster, which further merge to create a new cluster, and so on to create the above clusters(in green and red). These clusters further merge to create 1 cluster which contains all the data (blue line).

Based on the above dendrogram, and taking the distances into consideration, we will cut the dendrogram at distance = 15, thereby dividing the data into 3 clusters.

We get 3 clusters, which can be divided into categories, **high vulnerability, medium vulnerability, low vulnerability** with 96, 104 and 100 observations each.

4. Apply K-Means clustering on scaled data

In K- Means clustering, we need to identify the number of clusters, i.e. k in which our dataset would be clustered. The algorithm identifies K number of Centroids and then allocates all data points to the nearest cluster. To identify the ideal number of k, we calculate the inertia of varied values of k i.e. within cluster sum of squares and plot an elbow curve.



A low inertia value is preferred as the total intra cluster variance is minimized, which would make the cluster compact. Looking at the elbow curve, we should select a value of k, where inertia is low and also there is not much change in the inertia value with increasing k. We get the elbow like curve at 3, so we will take k =3, though we also have an elbow at 4, but the elbow at 3 is sharper.

The Silhouette Coefficient is calculated using the mean intra-cluster distance (a) and the mean nearest-cluster distance (b) for each sample. The Silhouette Coefficient for a sample is (b - a) /max(a, b)

Silhouette score (for k=3) = 0.535

5. Recommend different priority based actions.

Hierarchical Clustering

Below table gives the average value across all features for each cluster.

According to Hierarchical Clustering, 96 observations categorized in cluster 2 having lowest Health_Indeces1, Health_indices2, per_capita_income and GDP have priority 1 vulnerability, followed by cluster 3. Government should focus on cluster 2 states and improve health conditions.

#	Average				#	Categories
Cluster	Health_indeces1	Health_indices2	Per_capita_income	GDP	# Obs	
2	**400**	**104**	**680**	**5400**	**96**	**High Vulnerability**
3	2480	750	2350	136000	104	Medium Vulnerability
1	5000	1200	3375	377132	100	Low Vulnerability

K- Means Clustering

#	Average				#	Categories
Cluster	Health_indeces1	Health_indices2	Per_capita_income	GDP	# obs	
1	**500**	**116**	**693**	**9428**	**102**	**High Vulnerability**
3	2597	783	2500	140000	102	Medium Vulnerability
2	4930	1212	3385	380000	96	Low Vulnerability

According to K means Clustering, 102 observations categorized in cluster 1 having lowest Health_Indeces1 , Health_indices2 , per_capita_income and GDP have priority 1 vulnerability, followed by cluster 3. Government should focus on cluster 1 states and improve health conditions.