Comparing GDP per capita with energy use and co2 production

➢There can be no economic expansion or development without a significant influence on the environment due to the release of carbon dioxide from the production, transportation, and use of energy. As a result, climate change is primarily driven by the release of carbon dioxide.

Clustering of GDP per capita with energy use and co2 production

➢I have used clustering methods on a climate dataset I got from World Bank Open Data. The interesting clusters I found are co2, GDP and energy.

Fig.1 – Technologies used

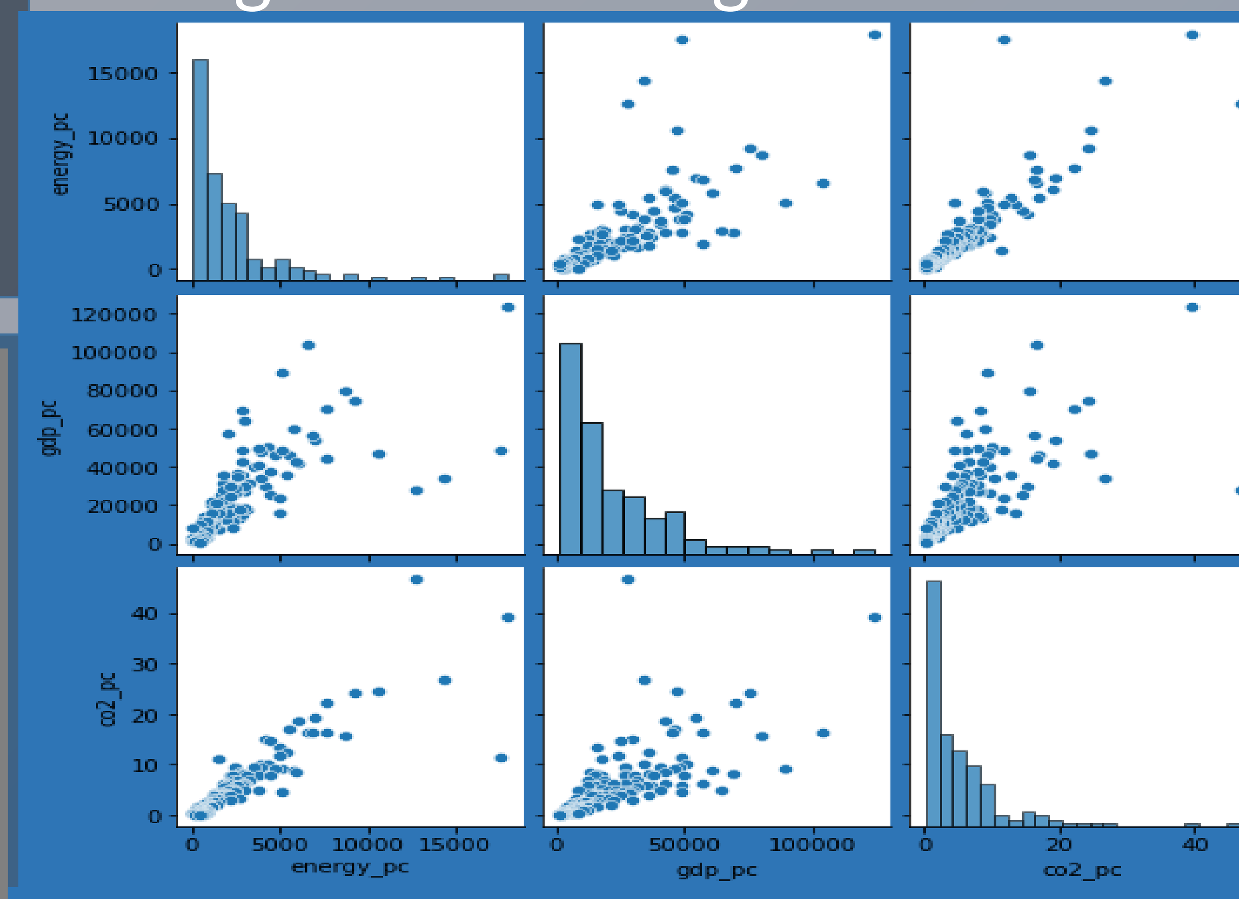


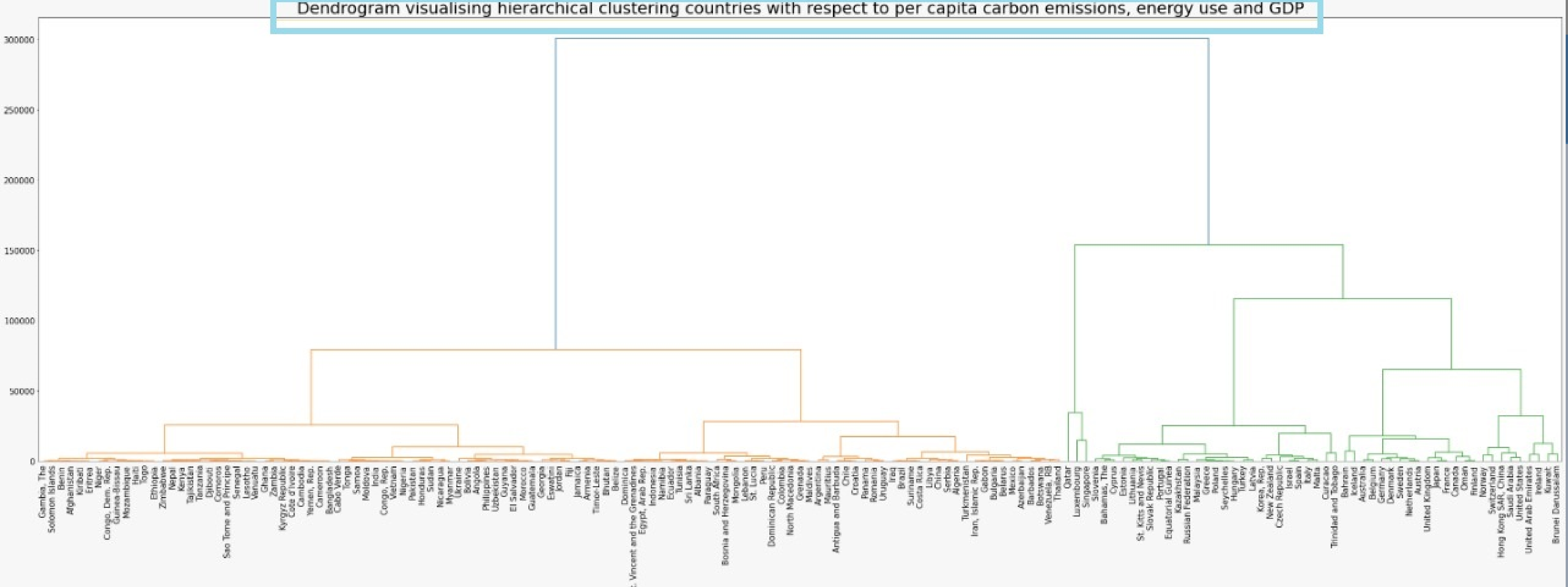
Fig.2 -Visuals of Relationships]

Exploring the data from World development indicators:

- I used pandas to explore the dataset.

Country Name	Country Code	Energy use (kg of oil equivalent per capita) 2015	GDP per capita, PPP (current international $) 2015	CO2 per capita (ton CO2/cap) 2015

- The year chosen for the dataset is the year 2015
- For plotting I had to import matplotlib.pyplot as plt, seaborn and numpy.



Dendrogram visualizing hierarchical clustering countries

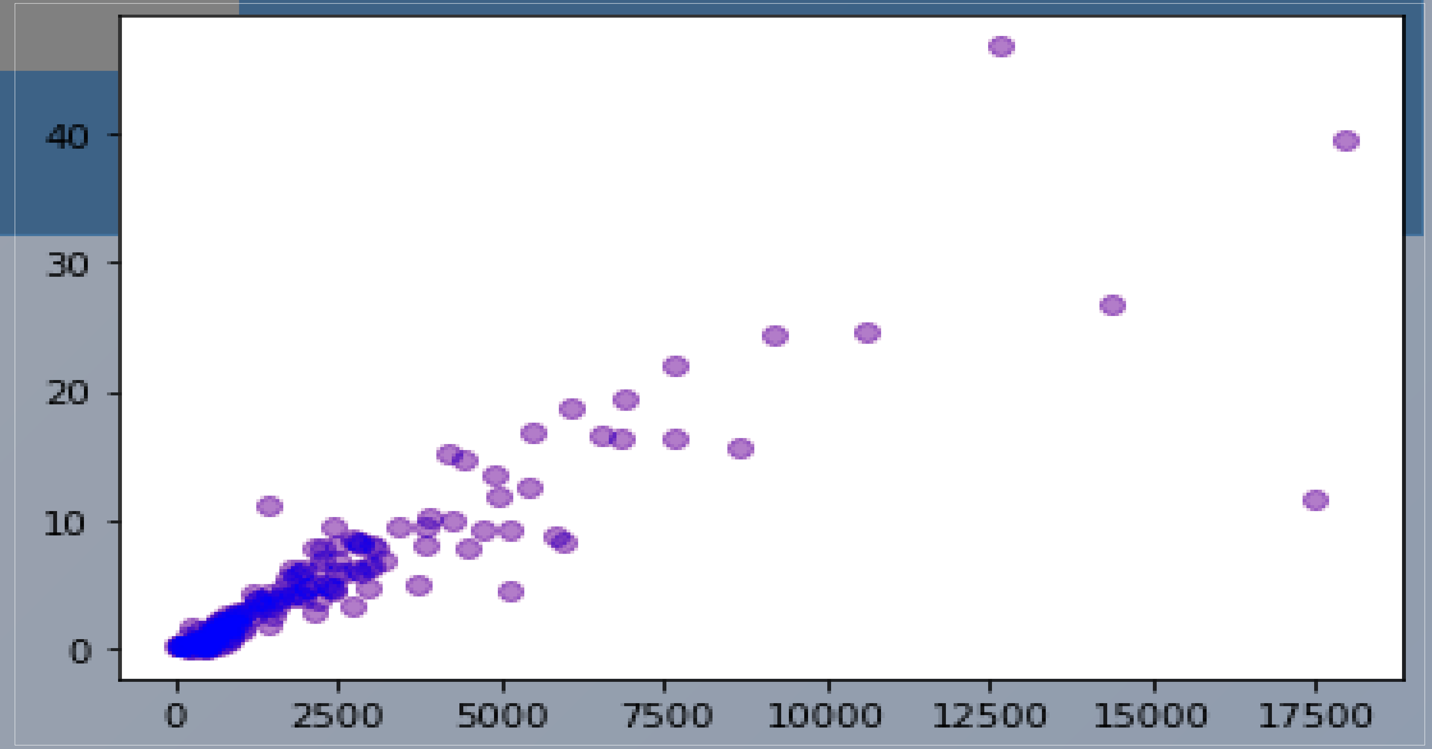


Fig.3- Energy_pc relationship with co2_pc

Code to get minimum and maximum normalization

➢Min max scaler Transforms features by scaling each feature to a given range.

➢Below is the code to get min and max normalization.

```
from sklearn import preprocessing

std_scale = preprocessing.StandardScaler().fit(climate_df[['energy_pc', 'gdp_pc', 'co2_pc']])
std = std_scale.transform(climate_df[['energy_pc', 'gdp_pc', 'co2_pc']])
climate_df_std = pd.DataFrame(data = std)

minmax_scale = preprocessing.MinMaxScaler().fit(climate_df[['energy_pc', 'gdp_pc', 'co2_pc']])
minmax = minmax_scale.transform(climate_df[['energy_pc', 'gdp_pc', 'co2_pc']])
climate_df_minmax = pd.DataFrame(data = minmax)
```

Clustering:

❑**Here, Using Kmeans we got 3 clusters which are partitioned using code which are shown in Fig. 4.**

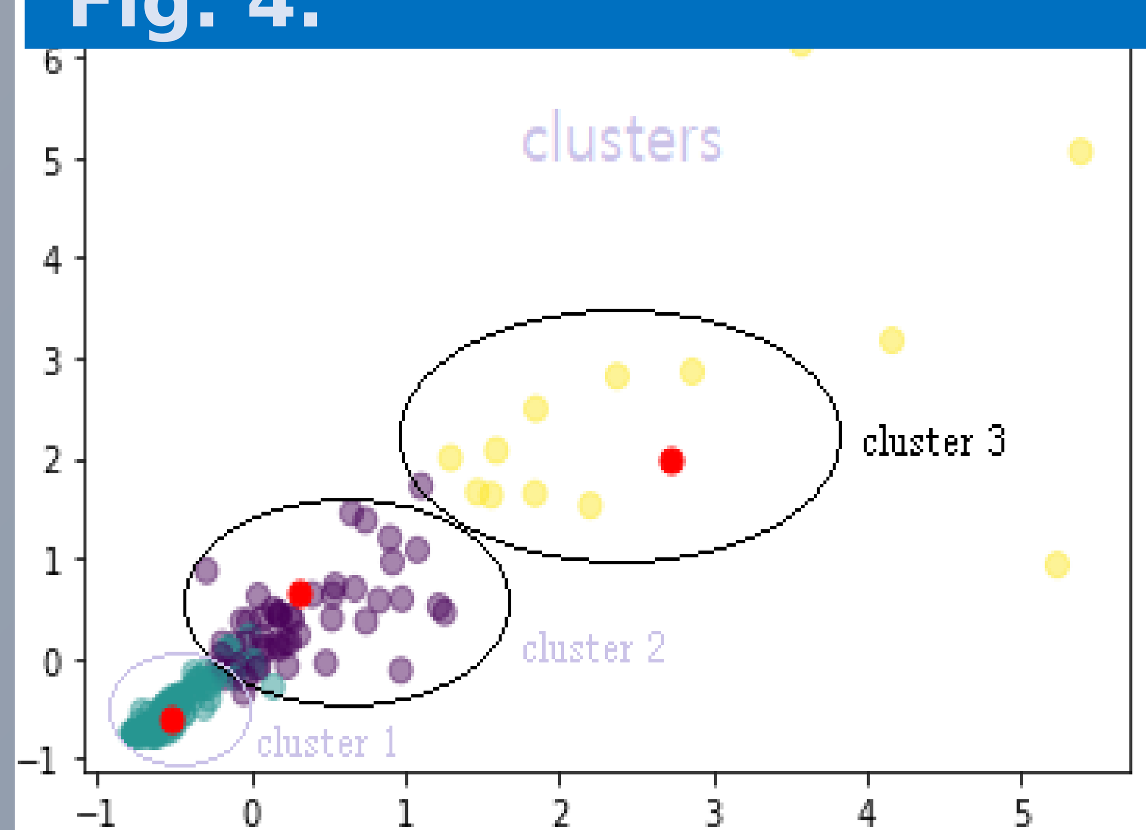


Fig.4 Clusters partitioning

Optimising the No. of Clusters

✓Following is the code to optimise the number of clusters by using Elbow method.

```
inertias = []
ks = range(1,8)

for k in ks:
    model = KMeans(n_clusters=k, init='k-means++', n_init=50).fit(climate_df_std)
    inertias.append(model.inertia_)

plt.plot(ks, inertias, '-o')
plt.xlabel('number of clusters')
plt.ylabel('inertia value')
plt.title('Investigation of optimal n clusters with elbow method')
```

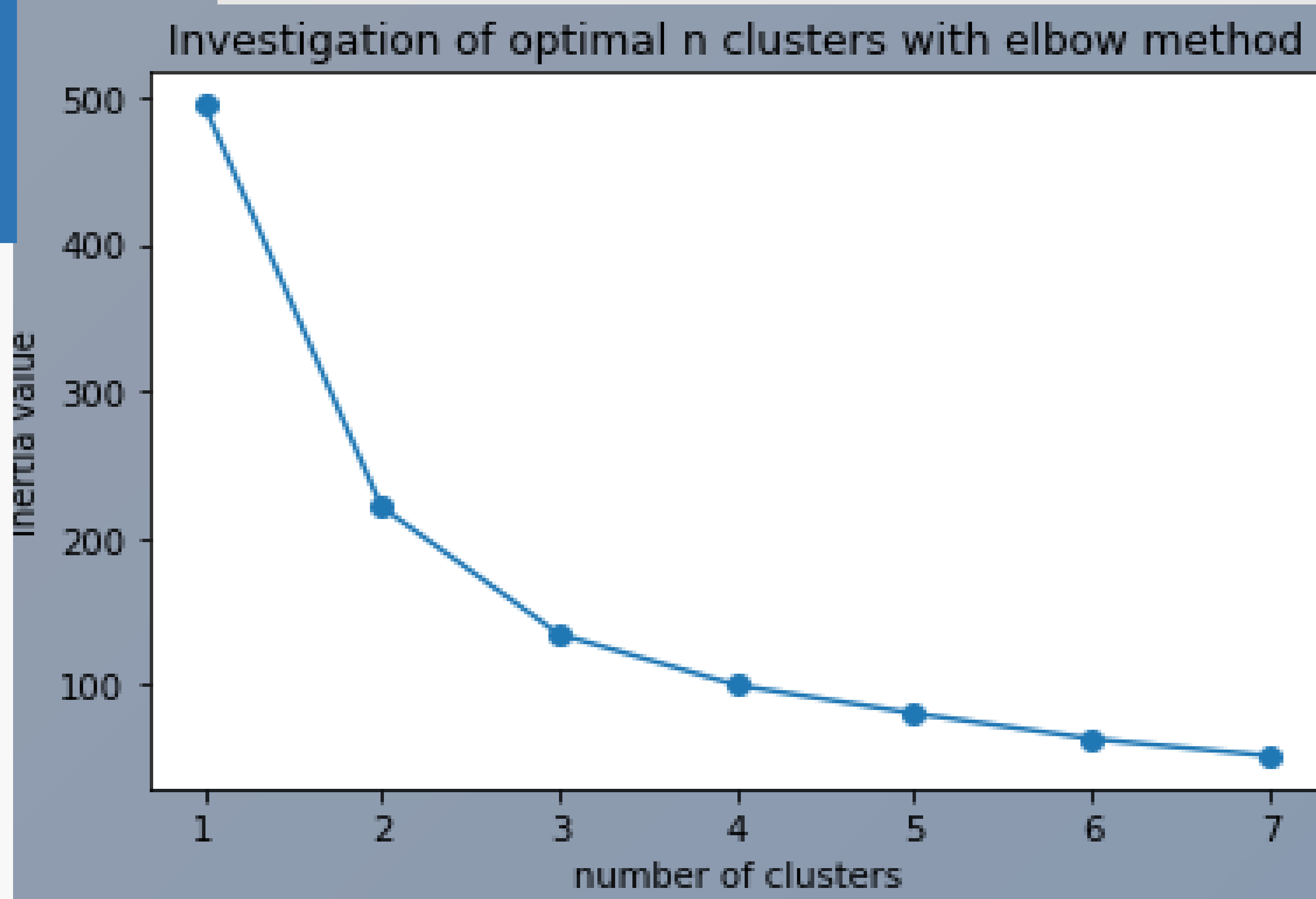


Fig.5 - Curve Fit with the number of clusters

Graphs:

❖The Second graph depicts hierarchical clustering using the ward technique and a dendrogram, which is labeled. Due to large number of data the dendrogram is blurred but still it's a good visualization.

❖In the Fig.5,we created simple model(s) fitting data sets with curve fit.

Poster by Priya Reddy Vadde