ENMGT 5930 - Data Analytics Course Project 1

Yatin Satija	ys2347
Rashmi Cherukuru	rc943
Yash Deshmukh	yvd2
Unnati Deshwal	ud37

Introduction

The need for a global transition from fossil fuels to renewable energy sources has never been more urgent. As countries strive to meet sustainability targets, determining the optimal pathway for renewable energy adoption becomes a critical challenge. Various factors, including economic conditions, resource availability, and current energy usage patterns, influence each country's unique path to sustainable energy. By analyzing historical energy consumption data, nuclear adoption rates, and other relevant metrics, we can apply clustering and predictive modeling techniques to identify optimal renewable adoption strategies tailored to different country profiles.

In this project, titled **"Energy Transition Pathways: Clustering and Predictive Analysis for Renewable Energy Insights,"** we aim to analyze global energy consumption patterns using clustering techniques and predictive modeling, identify distinct energy consumption profiles across countries, and provide insights into strategies for transitioning to renewable energy. The project focuses on segmenting countries based on their energy mix and consumption characteristics and understanding the factors influencing energy usage to inform global sustainability initiatives.

Our workflow encompasses the following methods:

Step-by-Step Flow of the Project

1. **Load and Preprocess Data:**
    - Import energy consumption data from two sources: "World Energy Consumption" and "Global Power Plant Database."
    - Select relevant features (e.g., fossil fuel consumption, nuclear, renewables consumption) for further analysis.
    - Perform feature engineering by calculating total_energy_consumption and percentage shares of renewables and nuclear energy within the total energy consumption.
    - Merge datasets on the country column and handle missing values to prepare the data for clustering and analysis.

2. **Scaling and Dimensionality Reduction:**
    - Apply StandardScaler to normalize numerical features (fossil fuel, nuclear, and renewables consumption).
    - Standardizing ensures the features are on a similar scale, which helps clustering algorithms to perform effectively.

3. **Determine Optimal Number of Clusters:**
    - Use the Elbow Method to examine the Sum of Squared Errors (SSE) for different numbers of clusters and identify the "elbow" point where adding more clusters does not significantly reduce SSE.
    - Apply the Silhouette Method to calculate silhouette scores, which measure how similar points in each cluster are. This helps validate the optimal cluster number.
    - Based on results, set an optimal number of clusters for further steps.

4. **Clustering with KMeans:**
   - Perform KMeans clustering using the selected number of clusters to group countries based on their energy consumption patterns.
   - Add the resulting cluster labels to the dataset for further analysis and visualization .
5. **PCA for Visualization:**
   - Apply Principal Component Analysis (PCA) to reduce the data's dimensionality to two principal components.
   - Plot the clustered data in 2D space using the PCA components to visually inspect the clusters and understand their separability.

6. Linear Regression for Energy Consumption Analysis:

- **Objective**: Develop a predictive model to understand factors influencing energy consumption and assess their relative impacts.
- **Feature Selection**:
  - Use key variables such as temperature, humidity, building square footage, occupancy, and renewable energy contributions.

This step-by-step approach combines clustering, time series forecasting, and reinforcement learning to analyze and optimize renewable energy strategies.

Datasets Overview

Data on Energy by Our World in Data

https://www.kaggle.com/datasets/pralabhpoudel/world-energy-consumption

This dataset, curated by Our World in Data, offers extensive metrics on global energy consumption, covering primary energy, per capita consumption, and growth rates. It also includes details on the energy mix and electricity mix, providing a holistic view of energy trends across countries. Updated regularly, this dataset supports robust analyses of global energy shifts and informs research on sustainable energy transitions.

Energy-consumption-prediction

https://www.kaggle.com/datasets/mrsimple07/energy-consumption-prediction

Designed for predictive modeling and experimentation, this synthetic dataset captures various environmental features, such as temperature, humidity, occupancy, HVAC usage, lighting, and renewable energy contributions. Each timestamp represents a snapshot of hypothetical energy consumption, enabling the study of factors influencing energy use. It provides a controlled environment to develop models and gain insights into energy consumption behaviors and patterns.

Nuclear Energy Dataset

https://www.kaggle.com/datasets/alistairking/nuclear-energy-datasets

This dataset on nuclear energy provides detailed data from reputable sources like the U.S. EIA, World Resources Institute, and Our World in Data. It includes information on nuclear power plant locations, uranium production, electricity generation, and safety records, enabling comprehensive analysis of nuclear energy trends, safety benchmarks, fuel dynamics, and the role of nuclear in global decarbonization. Potential expansions include granular plant-level data and policy information. The dataset is a valuable resource for research and informed discussions on nuclear energy's role in the energy transition and climate action.

Core Libraries and Functionalities

1. Scikit-learn:

- **StandardScaler and MinMaxScaler**: Used for feature scaling, ensuring data features have consistent ranges, which improves the performance and convergence of many machine learning algorithms.
- **KMeans**: A clustering algorithm that groups data points into clusters based on feature similarity, useful for segmenting data and discovering patterns.
- **PCA (Principal Component Analysis)**: A dimensionality reduction technique that reduces the number of features while preserving data variance, enhancing computational efficiency and reducing model complexity.
- **Mean Squared Error**: A regression metric that quantifies the average of squared differences between predicted and actual values, indicating the model's prediction accuracy.
- **Silhouette Score**: Measures the quality of clustering, evaluating how similar each point is to its cluster compared to others.

2. TensorFlow and Keras:

- **Sequential**: A Keras API model that allows building deep learning models layer by layer, suitable for sequential data processing tasks.
- **LSTM (Long Short-Term Memory)**: A specialized RNN layer used in time-series forecasting, particularly effective in handling temporal dependencies.
- **Dense**: A fully connected neural network layer, where each node connects to every other node in the previous layer.
- **Dropout**: A regularization layer that randomly drops units during training to prevent overfitting.

3. Stable Baselines3 (A2C): A reinforcement learning library that provides tools and algorithms for building, training, and deploying reinforcement learning models. A2C (Advantage Actor-Critic) is one of the RL algorithms available, suited for sequential decision-making problems in environments like those provided by OpenAI Gym.

4. gym: A toolkit for developing and comparing reinforcement learning algorithms, providing a variety of environments to test RL models.

5. Matplotlib (plt): A core plotting library in Python, essential for creating static, interactive, and animated visualizations. It provides control over every aspect of plot design.

6. Seaborn (sns): A data visualization library built on Matplotlib that makes it easier to create aesthetically pleasing statistical graphics. Often used for plotting distribution, correlation, and categorical data.

7. Plotly Express (px): A high-level, interactive data visualization library for quickly creating plots, allowing for easy customization and rendering of visualizations that are both visually appealing and informative.

Data Preparation and Transformation (Pre-Processing)

1. **Feature Engineering -** Feature engineering is the process of selecting and creating features from raw data to improve the performance of machine learning models.

    a. **Selecting Relevant Columns**

```
energy_data = world_energy_df[['country', 'year','fossil_fuel_consumption',
'nuclear_consumption', 'renewables_consumption']]
nuclear_data = nuclear_energy_df[['country', 'capacity_mw','primary_fuel']]
```

- The code extracts specific columns of interest from two datasets:
    - **world_energy_df,** which contains energy consumption data for fossil fuels, nuclear energy, and renewables.
    - **nuclear_energy_df** contains data about nuclear power plants, including their capacity and primary fuel type.
- These columns are chosen because they are relevant for understanding energy consumption patterns.

    b. **Calculating Total Energy Consumption**

```
energy_data['total_energy_consumption'] = energy_data[['fossil_fuel_consumption',
'nuclear_consumption', 'renewables_consumption']].sum(axis=1)
```

- A new feature, total_energy_consumption, is derived by summing up the consumption of fossil fuels, nuclear energy, and renewables for each country and year.
- This gives an overall picture of how much energy is consumed in total.

c. **Calculating Proportional Energy Shares**

```
energy_data['renewables_share_elec']= energy_data['renewables_consumption'] /
energy_data['total_energy_consumption']
energy_data['nuclear_share_energy'] = energy_data['nuclear_consumption'] /
energy_data['total_energy_consumption']
```

- Two new features are created:
  - renewables_share_elec: The share of energy derived from renewables as a fraction of total energy consumption.
  - nuclear_share_energy: The share of energy derived from nuclear sources as a fraction of total energy consumption.
- These proportional features are crucial for clustering because they capture the relative importance of different energy sources.

d. **Merging Datasets**

```
data = pd.merge(energy_data, nuclear_data, on='country', how='left')
```

- The energy data and nuclear data are merged on the country column to combine information from both datasets.
- The merge is performed as a "left join," meaning that all rows from energy_data will be retained, and matching rows from nuclear_data will be added.

2. **Data Cleaning**

a. Dropping Rows with Missing Values

```
data.dropna(subset=['fossil_fuel_consumption', 'nuclear_consumption', 'renewables_consumption'],
inplace=True)
```

- Rows with missing values in key energy consumption columns are dropped to ensure data integrity.
- This step is essential because missing values could lead to incorrect calculations or errors during analysis.

3. **Data Scaling**

a. Standardizing Features

```
scaler = StandardScaler()
scaled_data = scaler.fit_transform(data[['fossil_fuel_consumption', 'nuclear_consumption',
'renewables_consumption', 'renewables_share_elec', 'nuclear_share_energy']])
```

- The selected features are scaled using **StandardScaler**:
  - **Purpose**: Scaling ensures all features have a mean of 0 and a standard deviation of 1, which is essential for clustering algorithms (like K-Means) that are sensitive to feature magnitudes.
  - **Features Scaled**:
    - Absolute consumption values: fossil_fuel_consumption, nuclear_consumption, renewables_consumption.
    - Proportional values: renewables_share_elec, nuclear_share_energy.

- The scaling step prepares the data for clustering by normalizing the different feature scales.

K-Means Clustering

K-Means Clustering is a popular unsupervised machine learning algorithm used for grouping data points into distinct clusters based on their similarities. In our project, we used K-Means to segment countries by their energy consumption profiles (fossil fuel, nuclear, renewable) and identify clusters that could inform tailored renewable energy adoption strategies.

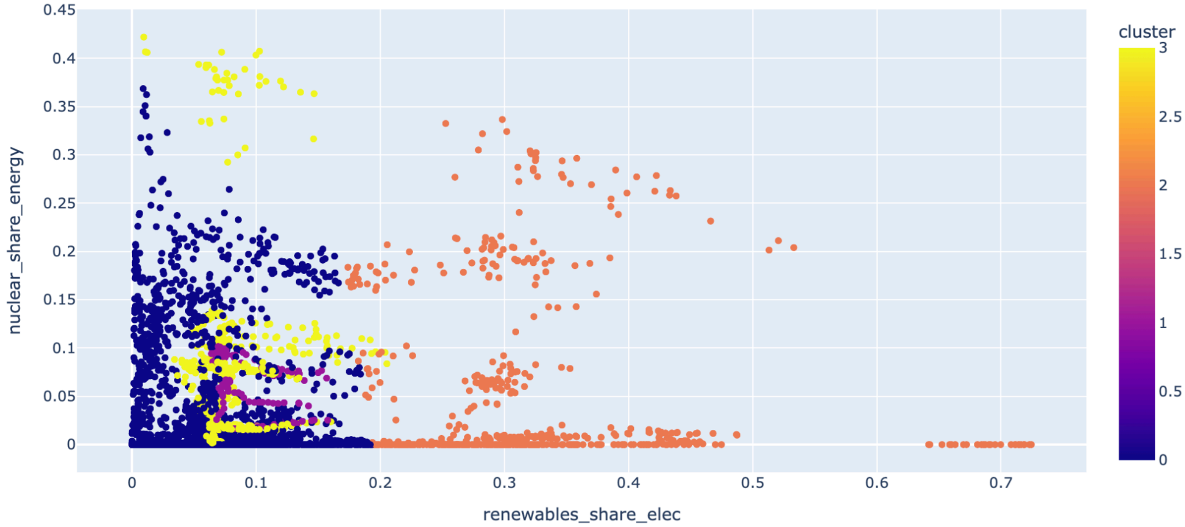
Initial Clustering with K-Means (n=4)

To start, we apply the K-Means clustering algorithm with n=4 clusters. However, the clusters were not well-defined, indicating that this number may not capture the underlying structure in the data accurately.

```
# Step 3: Clustering
kmeans = KMeans(n_clusters=4, random_state=42)
data['cluster'] = kmeans.fit_predict(scaled_data)
```

```
# Visualizing Clusters
fig = px.scatter(data, x='renewables_share_elec', y='nuclear_share_energy', color='cluster',
hover_data=['country'])
fig.show()
```

K-Means clustering is applied to group countries into 4 clusters based on their energy consumption patterns, using scaled features like fossil fuel, nuclear, and renewable shares. Each country is assigned a cluster label stored in the cluster column. An interactive scatter plot visualizes the clusters, with renewables_share_elec on the x-axis and nuclear_share_energy on the y-axis, allowing easy identification of grouping patterns and key trends.



Cluster 0 (Purple):

- **Characteristics:**
  - Countries in this cluster have very low shares of both renewable energy (renewables_share_elec) and nuclear energy (nuclear_share_energy).
  - Heavy reliance on fossil fuels for energy generation.

- **Insights:**
  - These countries might be underdeveloped or developing nations with minimal investment in clean energy technologies.
  - Likely candidates for energy transition initiatives.

Cluster 1 (Blue):

- **Characteristics:**
  - Countries have a low share of renewable energy but a slightly higher share of nuclear energy compared to Cluster 0.
  - Nuclear energy is playing a minor but noticeable role in these countries' energy portfolios.

- **Insights:**
  - These nations might have started investing in nuclear power but are still behind in renewable energy adoption.
  - There is scope to shift towards renewables.

Cluster 2 (Yellow):

- **Characteristics:**

    - Countries with a moderate to high share of nuclear energy and low to moderate shares of renewable energy.
    - A more balanced portfolio between nuclear and fossil fuels.

- **Insights:**
    - Likely to include countries with well-established nuclear infrastructure.
    - These countries might see nuclear energy as a transitional step while they gradually increase renewable adoption.

Cluster 3 (Orange):

- **Characteristics:**
    - Countries with a high share of renewable energy in electricity generation.
    - The share of nuclear energy is comparatively low.

- **Insights:**
    - These are advanced nations prioritizing renewable energy sources like solar, wind, and hydro.
    - They are leaders in the clean energy transition and have achieved significant progress in reducing fossil fuel dependency.

Overall Observations:

- **Trend 1:** Countries with high renewable shares tend to have lower nuclear shares and vice versa, indicating a preference for one type of clean energy strategy over the other.

- **Trend 2:** Clusters 0 and 1 show reliance on non-renewable sources, while Clusters 2 and 3 represent the transition to and leadership in clean energy technologies.

- **Energy Policy Implications:** Countries in Cluster 0 and Cluster 1 can focus on infrastructure development and policy changes to accelerate their renewable energy transition. Clusters 2 and 3 could set benchmarks and share best practices.

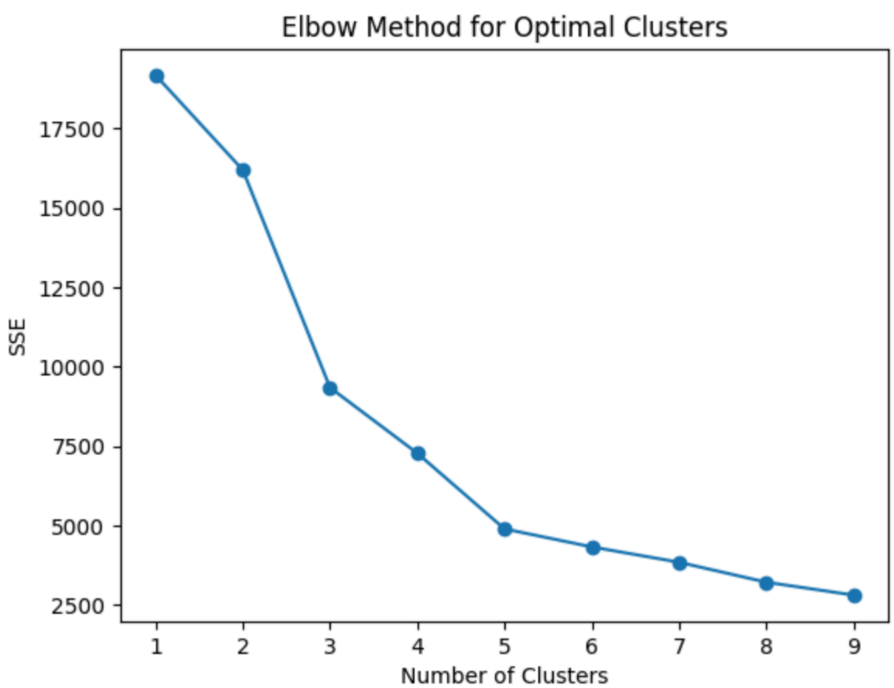
- **Lack of Clear Segmentation:** Clusters overlap significantly, making it hard to distinguish groups or derive actionable insights. One solution to solve this problem is determining the appropriate number of clusters.

Determining Optimal Number of Clusters

To refine our choice of clusters, we use the Elbow Method and Silhouette Method, two standard techniques for evaluating clustering quality:

Elbow Method: By plotting the sum of squared errors (SSE) for a range of cluster values, we identify the point where the improvement in SSE begins to level off, suggesting an optimal k.

```
# Elbow method
inertia = []
for i in range(1, 10):
    kmeans = KMeans(n_clusters=i, random_state=42)
    kmeans.fit(scaled_data)
    inertia.append(kmeans.inertia_)

plt.plot(range(1, 10), inertia, marker='o')
plt.xlabel('Number of clusters')
plt.ylabel('Inertia')
plt.title('Elbow Method')
plt.show()
```



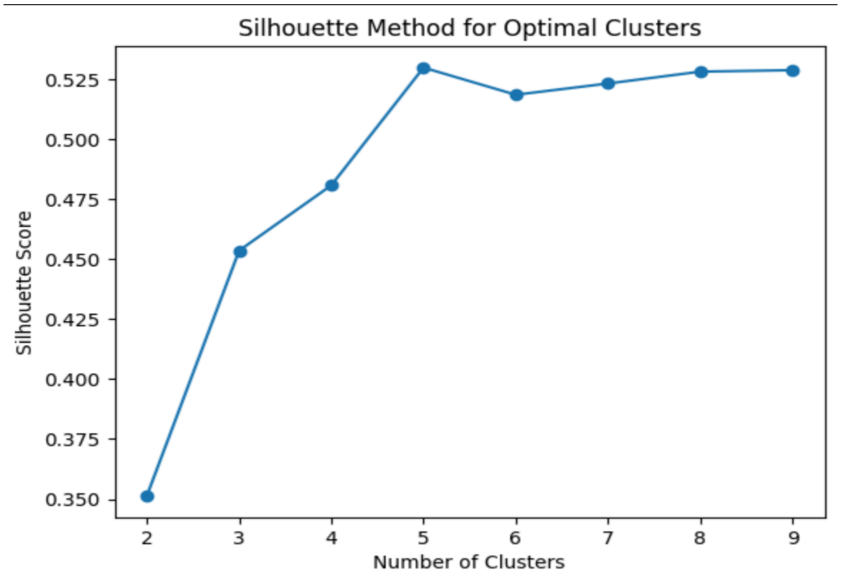
Based on the output the appropriate number of clusters should be 5. At **5 clusters**, the rate of decrease in inertia becomes more consistent, suggesting that splitting into more than 5 clusters might not provide substantial additional benefit.

<u>Silhouette Method</u>:- The method evaluates clustering quality by measuring how well data points fit within their assigned clusters compared to other clusters, with scores ranging from -1 to 1. A higher average silhouette score indicates better-defined clusters, helping to determine the optimal number of clusters.

```
# Silhouette Score
for i in range(2, 10):
  kmeans = KMeans(n_clusters=i, random_state=42)
  cluster_labels = kmeans.fit_predict(scaled_data)
  silhouette_avg = silhouette_score(scaled_data, cluster_labels)
  print(f"For n_clusters = {i}, the average silhouette score is : {silhouette_avg}")
```

OUTPUT

For n_clusters = 2, the average silhouette score is : 0.3513948222389523
For n_clusters = 3, the average silhouette score is : 0.4535077496770872
For n_clusters = 4, the average silhouette score is : 0.48073124198772144
For n_clusters = 5, the average silhouette score is : 0.5297654929159971
For n_clusters = 6, the average silhouette score is : 0.5184573851661148
For n_clusters = 7, the average silhouette score is : 0.5231388898866106
For n_clusters = 8, the average silhouette score is : 0.5280859912900308
For n_clusters = 9, the average silhouette score is : 0.5287008697841641

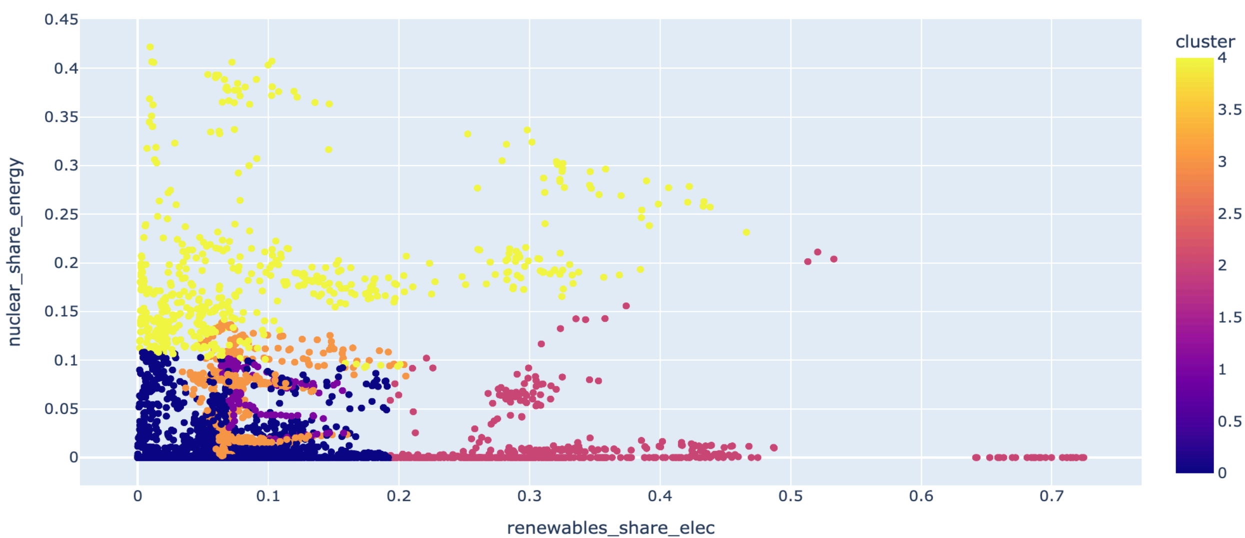


Based on the silhouette analysis, the optimal number of **clusters is 5**, as it achieves the highest average **silhouette score of 0.5298**, indicating well-defined and distinct clusters. Increasing the number of clusters beyond 5 results in marginal improvements or slight declines, suggesting diminishing returns.

K-Means Clustering (n=5)

```
kmeans = KMeans(n_clusters=5, random_state=42)
data['cluster'] = kmeans.fit_predict(scaled_data)
```

```
fig = px.scatter(data, x='renewables_share_elec', y='nuclear_share_energy', color='cluster',
hover_data=['country'])
fig.show()
```



Cluster 0 (Dark Blue):

- Represents countries with very low shares of both renewables and nuclear energy.
- Likely includes regions predominantly dependent on fossil fuels.
- This cluster remains compact and easy to identify.

Cluster 1 (Orange):

- Contains countries with moderate shares of renewables (approximately 0.1 to 0.2) and low nuclear energy.
- Represents regions transitioning to renewable energy sources but still with minimal nuclear reliance.

Cluster 2 (Pink):

- Comprises countries with very high renewable energy shares (>0.3) and negligible nuclear energy.

- Clear improvement in separation compared to the previous diagram, showing a distinct grouping of renewable leaders.

Cluster 3 (Yellow) :

- Represents countries with high nuclear shares (>0.2) and moderate renewable contributions.
- Improved separation from other clusters, highlighting nations with strong nuclear energy adoption.

Cluster 4 (Purple) :

- Includes countries with a balanced share of renewables and nuclear energy (moderate levels of both).
- Represents nations with a diversified energy mix.

Improvements from the Previous Diagram:

1. **Better Grouping of High Renewable Countries:**
   - Cluster 2 (high renewable share) is now better distinguished from others, particularly those with low nuclear energy reliance, compared to the earlier results.
2. **Improved Identification of Nuclear-Driven Economies:**
   - Cluster 3, with high nuclear share, is more clearly separated, showing improved grouping of nuclear-reliant nations.
3. **Addition of a New Balanced Cluster:**
   - Cluster 4 captures countries with a mixed energy portfolio (moderate shares of renewables and nuclear energy), adding a new layer of insight.

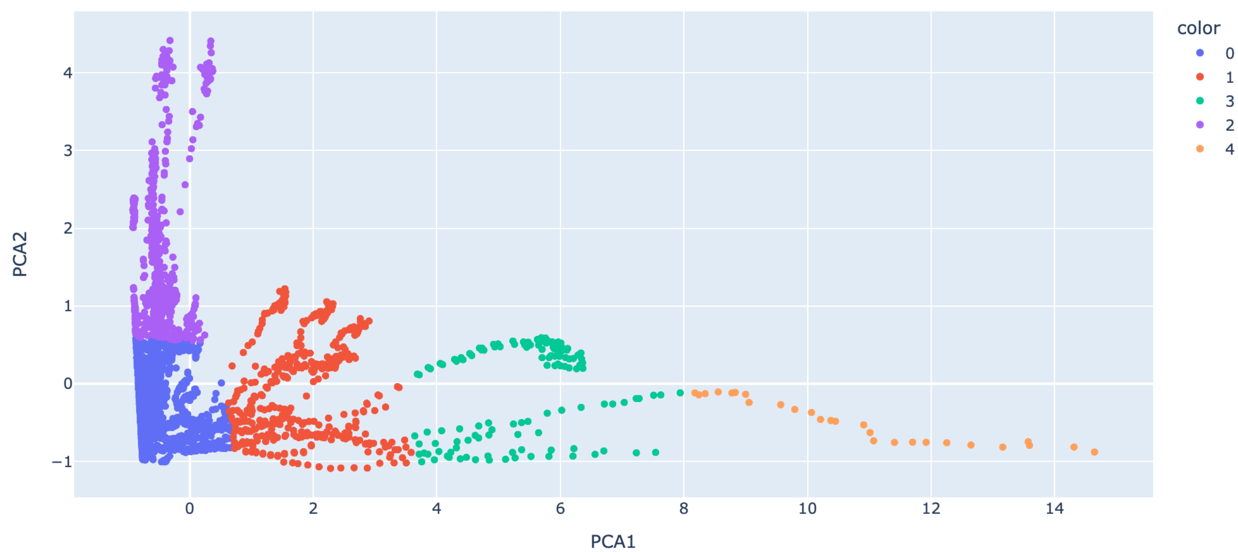
Remaining Challenges:

1. **Overlap Among Clusters:**
   - There is still significant overlap between Clusters 0, 1, and parts of 3, indicating a lack of clear separation in some regions of the diagram.
2. **Cluster Boundaries Are Not Sharp:**
   - The lack of distinct segmentation makes it challenging to report clear, non-overlapping categories, particularly for countries transitioning energy sources.

Enhancing Clustering with Principal Component Analysis (PCA)

Principal Component Analysis (PCA) is a dimensionality reduction technique that improves clustering performance by reducing data complexity. By identifying key features that capture the most variance in the data, PCA helps to remove noise and irrelevant features, making clusters more distinct. In combination with KMeans clustering, PCA reduces computational requirements and often enhances cluster separation, leading to more accurate and efficient clustering results.

```
from sklearn.decomposition import PCA
import pandas as pd
import plotly.express as px
from sklearn.cluster import KMeans

# Apply PCA
pca = PCA(n_components=2)
pca_data = pca.fit_transform(scaled_data)

pca_df = pd.DataFrame(pca_data, columns=['PCA1', 'PCA2'])
pca_df['country'] = data['country']  # Adding the country column
pca_df['cluster'] = KMeans(n_clusters=5, random_state=42).fit_predict(pca_data)

# Visualize the clusters
fig = px.scatter(pca_df, x='PCA1', y='PCA2', color=pca_df['cluster'].astype(str), hover_data=['country'])
fig.show()
```



Let’s analyze PCA1 and PCA2.

```
feature_names = ['fossil_fuel_consumption', 'nuclear_consumption', 'renewables_consumption',
'renewables_share_elec', 'nuclear_share_energy']
loadings_df = pd.DataFrame(pca.components_, columns=feature_names, index=['PCA1', 'PCA2'])
print(loadings_df)
```

OUTPUT

```
      fossil_fuel_consumption  nuclear_consumption  renewables_consumption  \
 PCA1                 0.587973             0.558149                0.578698
 PCA2                -0.122006             0.094553               -0.046962

      renewables_share_elec  nuclear_share_energy
 PCA1             -0.039528              0.079402
 PCA2              0.518973              0.839426
```

PCA1 Interpretation:

PCA1 has the following loadings:

- **Fossil Fuel Consumption (0.588)**: This feature contributes the most to PCA1, with a high positive influence.
- **Nuclear Consumption (0.558)**: Significant positive influence, slightly lower than fossil fuel consumption.
- **Renewables Consumption (0.579)**: Also contributes positively, on par with fossil fuel consumption and nuclear consumption.
- **Renewables Share in Electricity (-0.040)**: Negligible influence, with a slightly negative contribution.
- **Nuclear Share in Energy (0.079)**: Very low positive contribution.

Conclusion for PCA1:

- PCA1 is primarily a composite of **fossil fuel consumption**, **nuclear consumption**, and **renewables consumption**.
- It reflects a **general energy consumption trend**, focusing on absolute energy usage, irrespective of renewables or nuclear shares.

PCA2 Interpretation:

PCA2 has the following loadings:

- **Fossil Fuel Consumption (-0.122):** Minor negative contribution.
- **Nuclear Consumption (0.095):** Minor positive contribution.
- **Renewables Consumption (-0.047):** Negligible negative contribution.
- **Renewables Share in Electricity (0.519):** Significant positive contribution.
- **Nuclear Share in Energy (0.839):** Strongest positive contribution.

Conclusion for PCA2:

- PCA2 focuses on **energy composition**, specifically highlighting the shares of **renewables in electricity** and **nuclear in total energy**.
- It emphasizes the relative importance of these energy sources rather than their absolute consumption.

Learnings about the PCA-induced clusters

1. **Cluster 0 (Blue Points):**
    - **Characteristics:**
        - High contribution to PCA1 and low contribution to PCA2.
        - Represents entities (e.g., countries or regions) with high absolute energy consumption, especially dominated by **fossil fuels, nuclear, and renewables consumption**.
    - **Insights:** These are likely high-energy-consuming regions with less emphasis on renewables' share in electricity or nuclear's share in energy.

2. **Cluster 1 (Dark Orange Points):**
    - **Characteristics:**
        - Moderate-to-high PCA1 and moderate PCA2 values.
        - Represents regions with a balanced mix of energy consumption and a reasonable contribution from nuclear and renewables.
    - **Insights:** These regions may be transitioning toward a more balanced energy mix, combining absolute consumption with growing shares of renewables or nuclear.

3. **Cluster 2 (Purple Points):**
    - **Characteristics:**
        - Low to moderate PCA1 and high PCA2 values.
        - Indicates regions with lower overall energy consumption but a significant focus on **nuclear and renewables' shares** in their energy mix.
    - **Insights:** Likely to represent regions with advanced energy structures emphasizing sustainability (higher nuclear and renewable shares).

4. **Cluster 3 (Green Points):**
    - **Characteristics:**
        - Low PCA1 but moderate PCA2 values.
        - Represents entities with relatively lower overall energy consumption but some emphasis on renewables or nuclear shares.
    - **Insights:** These are possibly small or low-energy-consuming regions gradually adopting nuclear or renewable technologies.

5. **Cluster 4 (Orange Points on the Right):**
    - **Characteristics:**
        - High PCA1 and low PCA2 values, appearing distinct from the other clusters.
        - Represents regions with extreme energy consumption levels and minimal emphasis on renewable or nuclear shares.
    - **Insights:** These may include countries heavily dependent on **fossil fuels** or lagging in adopting sustainable energy practices.

DBSCAN Clustering

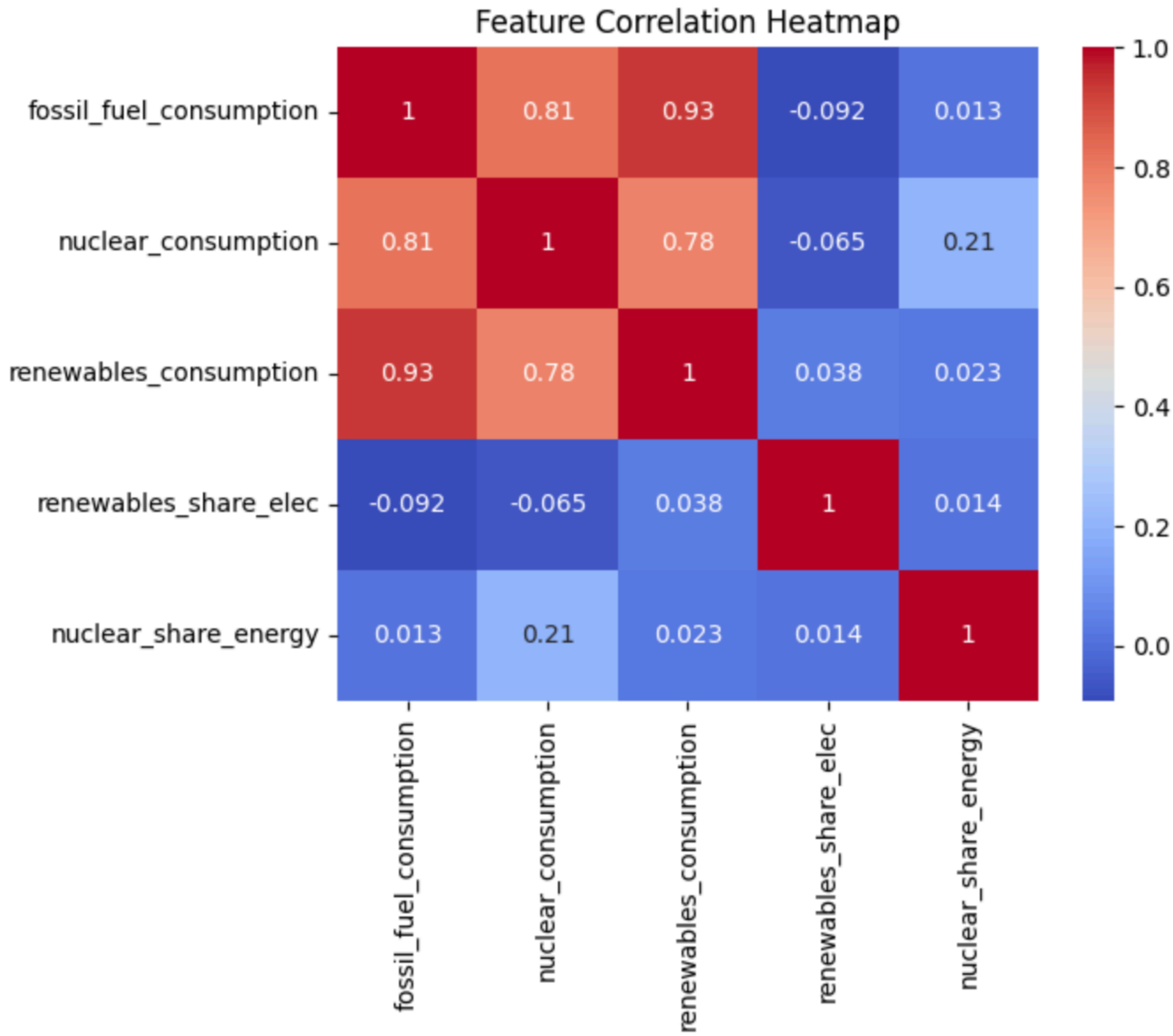
DBSCAN (Density-Based Spatial Clustering of Applications with Noise) is a clustering algorithm that groups points based on the density of data points in their vicinity. Unlike KMeans, DBSCAN does not require specifying the number of clusters in advance and can identify clusters of arbitrary shapes. It is particularly effective at handling noisy data by treating isolated points as outliers, making it well-suited for complex datasets with varying densities.

Step 1: Feature Selection

Feature selection is critical for improving clustering accuracy and interpretability. We followed these steps:

```
import seaborn as sns
import matplotlib.pyplot as plt

corr_matrix = data[['fossil_fuel_consumption', 'nuclear_consumption', 'renewables_consumption',
               'renewables_share_elec', 'nuclear_share_energy']].corr()
sns.heatmap(corr_matrix, annot=True, cmap='coolwarm')
plt.title('Feature Correlation Heatmap')
plt.show()
```

1. **Correlation Analysis**:
   - We plotted a heatmap to analyze feature correlations.
   - Features with strong correlations to energy consumption and renewable share metrics were identified as candidates for clustering.
2. **Selected Features**: Based on the correlation heatmap, we selected the following features for DBSCAN:
   - Renewables Consumption
   - Renewables Share in Electricity
   - Nuclear Share in Energy
3. These features capture key aspects of energy distribution and usage, ensuring domain relevance for clustering.



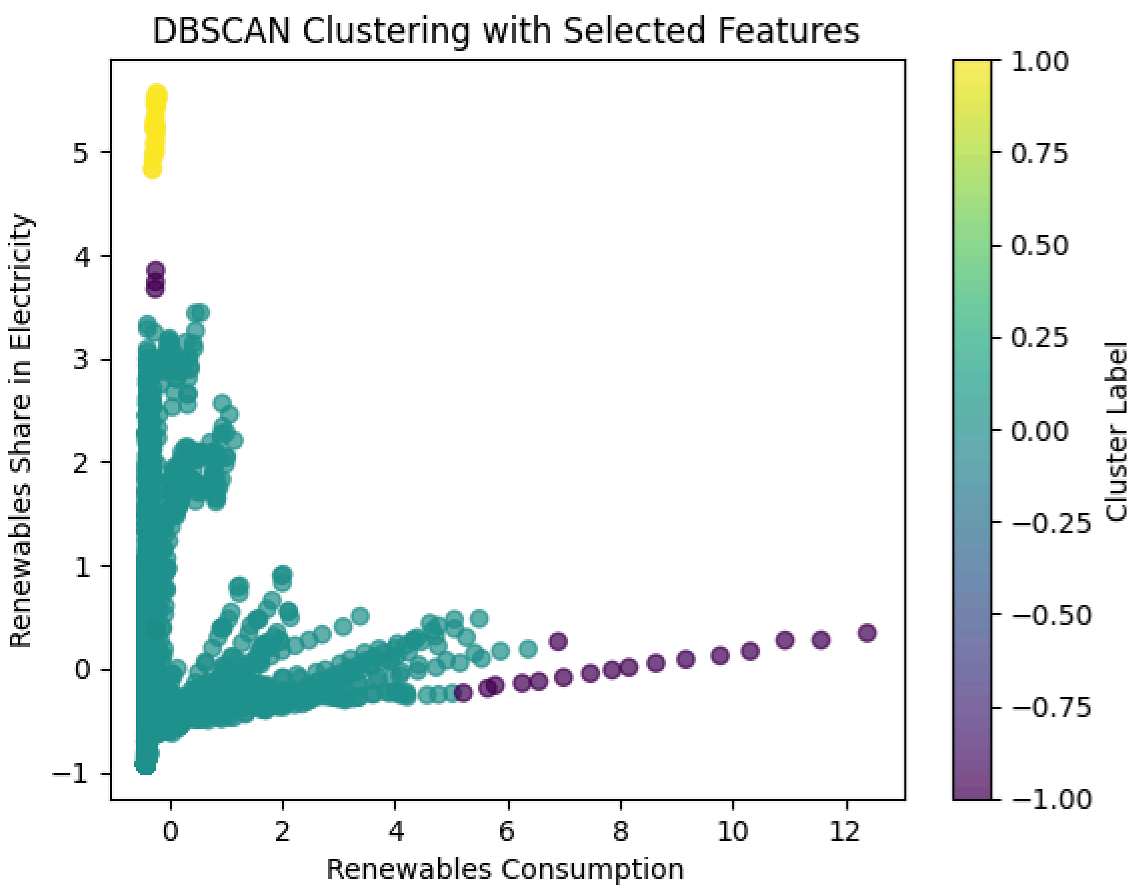
Step 2: DBSCAN Clustering

```
import pandas as pd
from sklearn.preprocessing import StandardScaler

# Assuming 'df' is your original DataFrame with the raw data
features = ['fossil_fuel_consumption', 'nuclear_consumption', 'renewables_consumption',
        'renewables_share_elec', 'nuclear_share_energy']

# Scale the data
scaler = StandardScaler()
scaled_data = scaler.fit_transform(data[features])
```

```python
# Convert scaled_data to a DataFrame
scaled_df = pd.DataFrame(scaled_data, columns=features)

# Selecting features for DBSCAN
selected_features = scaled_df[['renewables_consumption', 'renewables_share_elec',
'nuclear_share_energy']]

# Apply DBSCAN
from sklearn.cluster import DBSCAN
dbscan = DBSCAN(eps=0.5, min_samples=5)
clusters = dbscan.fit_predict(selected_features)

# Add clusters to the DataFrame
scaled_df['dbscan_cluster'] = clusters

# Visualize the clustering results
import matplotlib.pyplot as plt

plt.scatter(selected_features.iloc[:, 0], selected_features.iloc[:, 1],
        c=clusters, cmap='viridis', alpha=0.7)
plt.xlabel('Renewables Consumption')
plt.ylabel('Renewables Share in Electricity')
plt.title('DBSCAN Clustering with Selected Features')
plt.colorbar(label='Cluster Label')
plt.show()
```

1. **DBSCAN Algorithm**:
    - DBSCAN groups data points based on density, identifying core points and labeling sparse points as noise.
    - Parameters used:
        - eps = 0.5: Defines the neighborhood radius for density.
        - min_samples = 5: Minimum points required to form a dense region.

2. **Application**:
    - The algorithm was applied to the scaled dataset of selected features.
    - Noise points (cluster label -1) were separated, and dense regions were grouped into clusters.



1. **Yellow Cluster (High Outliers)**:
   - Represents regions with exceptionally high renewables share in electricity.
   - Likely leaders in renewable energy adoption due to strong policies or natural advantages.
   - Insight: Study these to identify strategies for accelerating renewable adoption elsewhere.

2. **Purple Cluster (Noise/Isolated Points)**:
   - Scattered points classified as noise, indicating irregular patterns in renewables data.
   - May represent regions with unstable energy policies or incomplete data.
   - Insight: Investigate reasons for their irregularity and address data gaps.

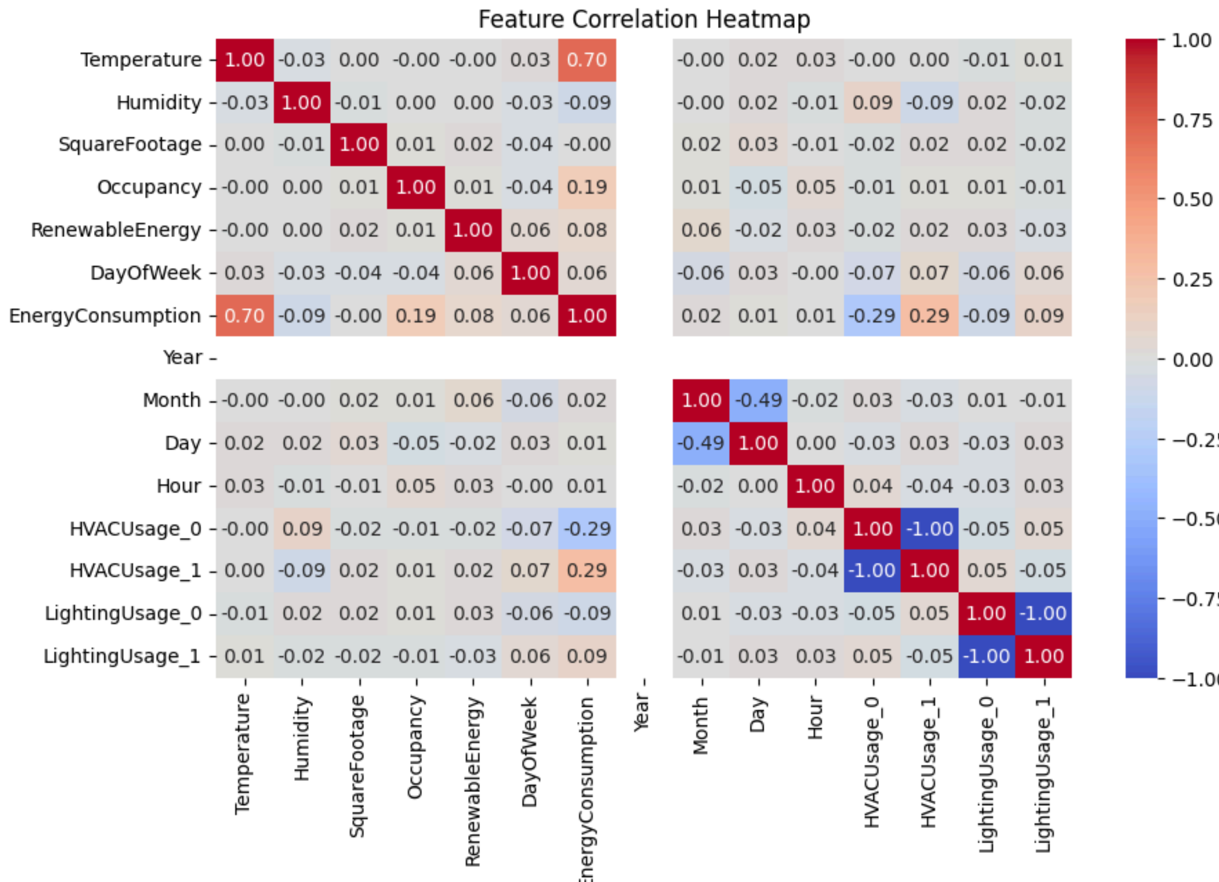
3. **Green Cluster (Main Group)**:
   - Majority of regions with moderate renewables adoption, reflecting a global transition trend.
   - Likely includes both developing and developed nations at various stages of the energy shift.
   - Insight: Analyze sub-trends to identify best practices or common challenges.

Linear Regression Analysis for Energy Consumption Dataset

1. Correlation Heatmap Analysis

- **Objective**: To identify relationships between different variables in the dataset and understand which features are significantly correlated with energy consumption.

```
plt.figure(figsize=(10, 6))
correlation_matrix = energy_consumption_df.corr()
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', fmt=".2f")
plt.title("Correlation Heatmap for Energy Consumption Dataset")
plt.show()
```



Key Observations:

- **Temperature and Energy Consumption**:
  - High positive correlation (~0.70) indicates that as temperature increases, energy consumption also rises. This is likely due to the increased use of air conditioning or cooling systems in higher temperatures.

- **HVAC Usage (HVACUsage_0 and HVACUsage_1)**:
    - Strong positive and negative correlations within HVAC-related variables were observed. This is expected, as HVAC systems significantly impact energy consumption patterns.

- **Humidity**:
    - Weak negative correlation (-0.09) with energy consumption, indicating a negligible impact in this dataset.

- **Renewable Energy**:
    - Very low correlation with other variables, suggesting that renewable energy integration does not directly influence energy consumption or other features in this dataset.

- **Occupancy**:
    - Slight positive correlation (~0.19) with energy consumption. This reflects that higher occupancy levels marginally increase energy use.

Insights:

- Temperature is a critical variable influencing energy consumption and should be a focus in predictive modeling.

Linear Regression Model

Objective: To create a predictive model for energy consumption using the selected features.

```
# --------------------- Linear Regression on Energy Consumption Dataset ---------------------
# Preprocessing
energy_features = energy_consumption_df[["Temperature", "Humidity", "SquareFootage",
"Occupancy", "RenewableEnergy"]]
energy_target = energy_consumption_df["EnergyConsumption"]
# Splitting into train and test sets
X_train, X_test, y_train, y_test = train_test_split(energy_features, energy_target, test_size=0.2,
random_state=42)
# Linear Regression Model
linear_reg = LinearRegression()
linear_reg.fit(X_train, y_train)
# Predictions and evaluation
y_pred = linear_reg.predict(X_test)
mse_energy = mean_squared_error(y_test, y_pred)
r2_energy = r2_score(y_test, y_pred)
# Results
energy_model_results = {
  "Coefficients": linear_reg.coef_,
  "Intercept": linear_reg.intercept_,
  "Mean Squared Error": mse_energy,
```

```
  "R2 Score": r2_energy
}
# Displaying results
print("Energy Consumption Dataset Results:")
print(energy_model_results)
```

OUTPUT

```
Energy Consumption Dataset Results:
{'Coefficients': array([ 1.98538118e+00, -6.39092250e-02, -5.82431499e-04,  4.88579667e-01,
9.85460181e-02]), 'Intercept': 27.516620263645507, 'Mean Squared Error': 33.05972041231882,
'R2 Score': 0.4952704849998406}
```

Dependent Variable (Target Variable):

- **Energy Consumption**:
  - This is the variable that the model is attempting to predict.
  - It represents the total energy used in the building, typically measured in kilowatt-hours (kWh).

Independent Variables (Features):

These are the predictors used to estimate or explain variations in the dependent variable:

1. **Temperature**: Represents the external temperature, which directly influences HVAC operations and energy usage.

2. **Humidity**: Indicates the moisture in the air, which impacts heating, cooling, and ventilation needs.

3. **SquareFootage**: The size of the building, a key determinant of energy requirements.

4. **Occupancy**: The number of people in the building, which influences lighting, HVAC, and other energy needs.

5. **RenewableEnergy**: Contribution of renewable energy to the overall energy mix, which can offset total consumption.

Train-Test Split:

- Dataset split into 80% training and 20% testing for model evaluation.

Results:

Model Coefficients

- **Temperature (+1.985)**: Energy consumption increases by ~1.99 units for each one-unit rise in temperature, likely due to cooling needs.
- **Humidity (-0.064)**: A slight decrease in energy consumption with higher humidity, potentially due to reduced HVAC use.
- **SquareFootage (-0.0006)**: Minimal negative impact, suggesting other factors dominate energy usage.
- **Occupancy (+0.489)**: Energy consumption increases by ~0.49 units for each additional occupant due to lighting and HVAC use.
- **RenewableEnergy (+0.099)**: Small positive impact, indicating renewable energy supplements energy usage rather than replacing it.

Intercept (+27.52)

- Baseline energy consumption when all variables are zero (not practically relevant).

Model Evaluation

- **MSE (33.06)**: Moderate prediction errors based on the dataset scale.
- **R-Squared (0.495)**: The model explains 49.5% of the variance in energy consumption, showing moderate fit.

Key Takeaways

- Temperature and occupancy are the strongest predictors.
- Humidity and square footage have little influence in this dataset.

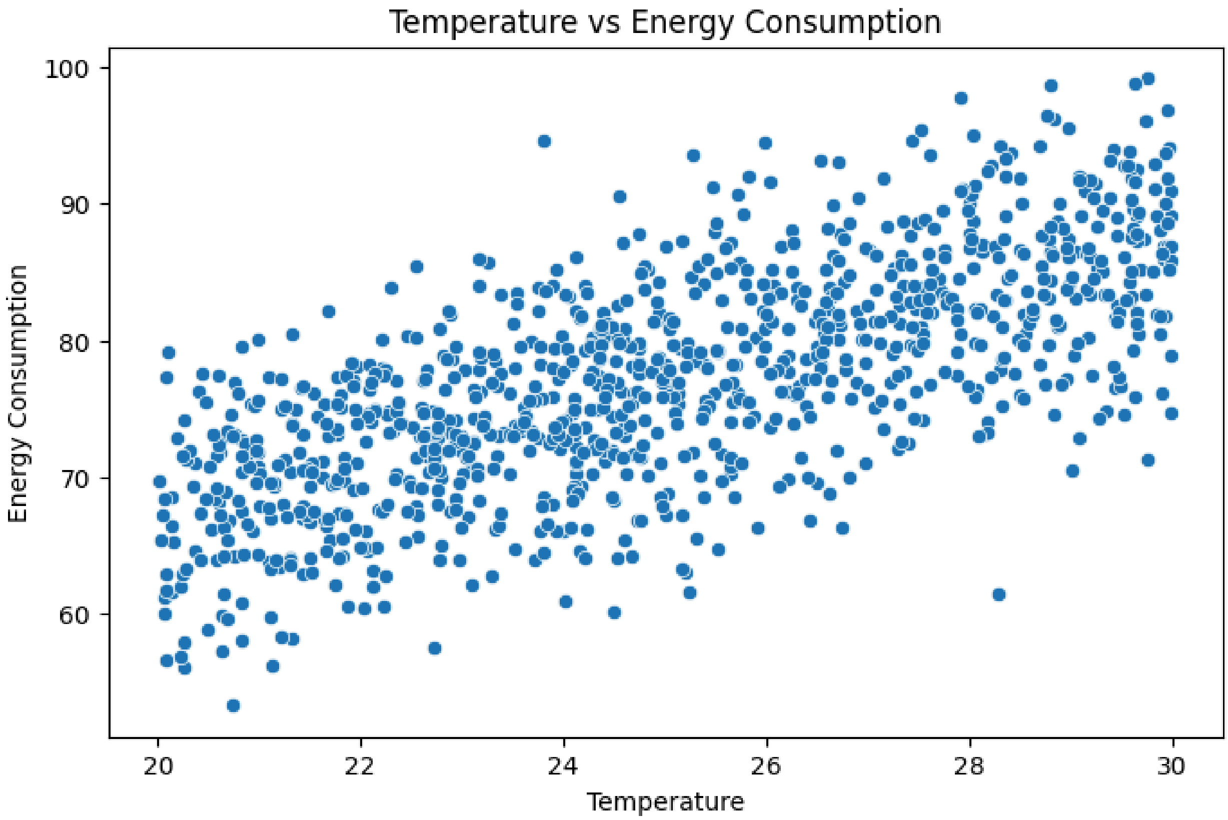
Scatter Plot Analysis

Objective: To visualize key relationships and evaluate model predictions.

Plots:

- **Temperature vs. Energy Consumption**:

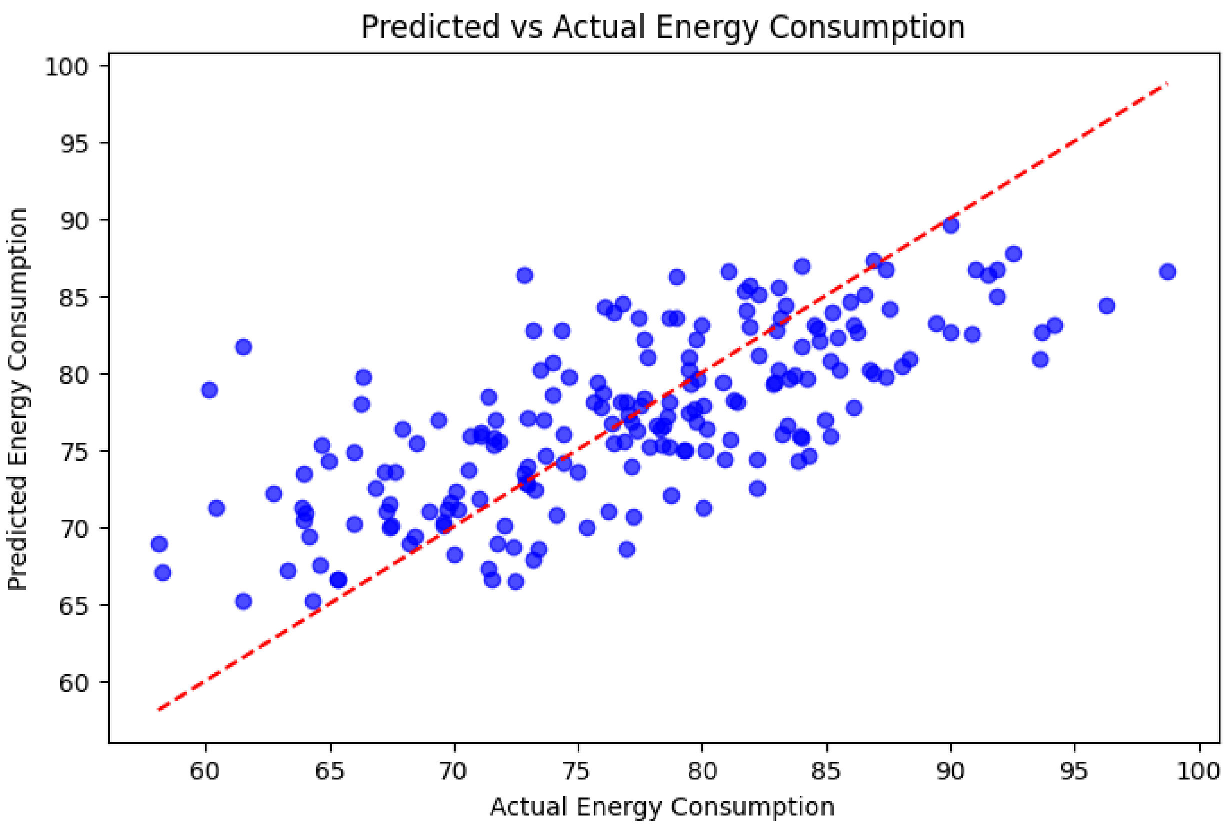
```
# Scatter plot for Temperature vs EnergyConsumption
plt.figure(figsize=(8, 5))
sns.scatterplot(x=energy_consumption_df["Temperature"],
y=energy_consumption_df["EnergyConsumption"])
plt.title("Temperature vs Energy Consumption")
plt.xlabel("Temperature")
plt.ylabel("Energy Consumption")
plt.show()
```



A scatter plot revealed a clear upward trend, confirming that higher temperatures are associated with increased energy consumption. This aligns with expectations due to temperature-sensitive cooling systems.

- **Predicted vs. Actual Energy Consumption**: A scatter plot compared model predictions to actual values.

```
# Scatter plot: Predicted vs Actual
plt.figure(figsize=(8, 5))
plt.scatter(y_test, y_pred, alpha=0.7, color='blue')
plt.title("Predicted vs Actual Energy Consumption")
plt.xlabel("Actual Energy Consumption")
plt.ylabel("Predicted Energy Consumption")
plt.plot([y_test.min(), y_test.max()], [y_test.min(), y_test.max()], 'k--', color='red')  # Diagonal line
plt.show()
```

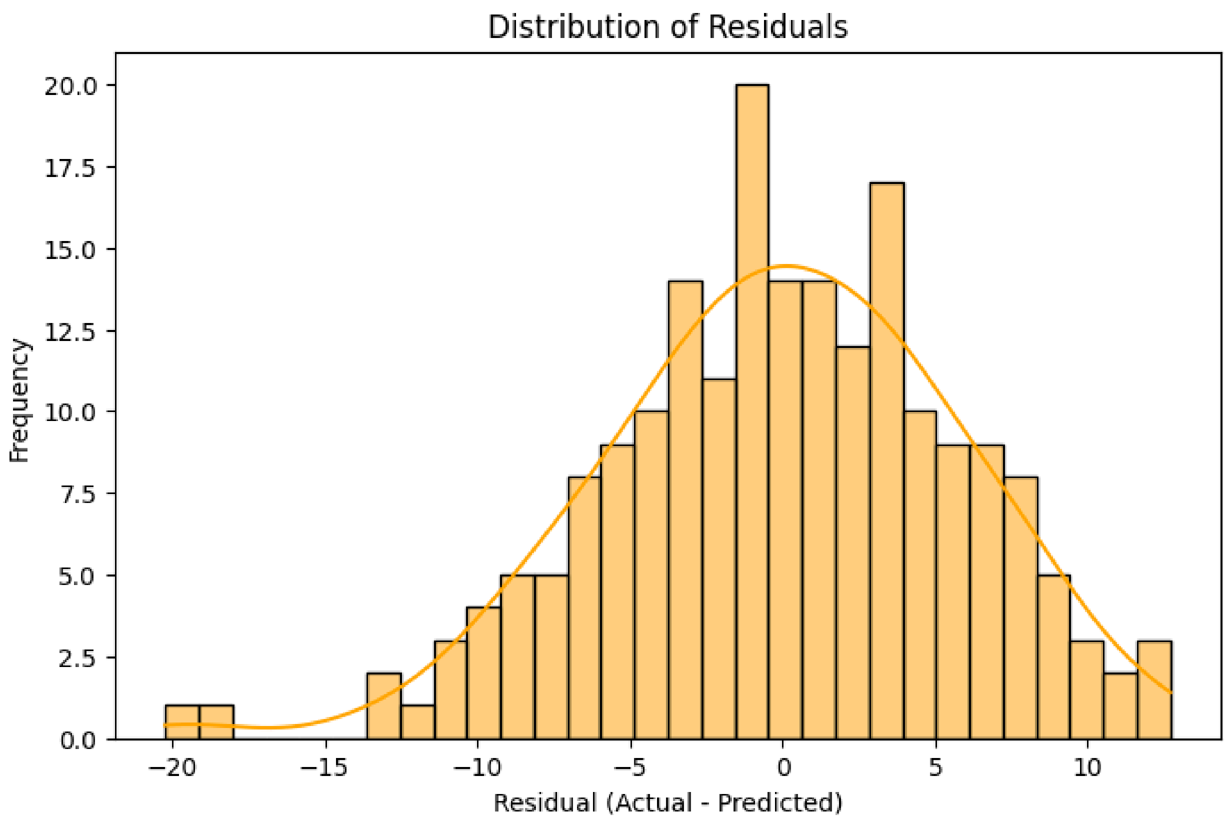


Data points closely cluster around the diagonal (perfect prediction) line, indicating that the model performs well. However, deviations from the line highlight areas where the model under- or over-predicts.

Residual Analysis

```
residuals = y_test - y_pred

plt.figure(figsize=(8, 5))
sns.histplot(residuals, kde=True, bins=30, color="orange")
plt.title("Distribution of Residuals")
plt.xlabel("Residual (Actual - Predicted)")
plt.ylabel("Frequency")
plt.show()
```

The residuals are centered around zero and approximately follow a normal distribution, indicating unbiased predictions and alignment with linear regression assumptions. The spread suggests moderate variability, with some outliers hinting at patterns not fully captured by the model.



Final Results and Conclusion

Final Results:

1. **Clustering Analysis**:
    - The KMeans algorithm identified distinct clusters of countries based on energy consumption patterns. Each cluster revealed unique profiles:
        - **Cluster 0**: High dependency on fossil fuels with minimal renewable or nuclear contributions.
        - **Cluster 1**: Moderate renewable shares but low nuclear energy reliance, indicative of countries in transition.
        - **Cluster 2**: High renewable adoption with negligible nuclear usage, representing advanced nations prioritizing sustainability.
        - **Cluster 3**: High nuclear energy usage with moderate renewable shares, likely nations with established nuclear infrastructure.
        - **Cluster 4**: Balanced contributions from both nuclear and renewables, showcasing a diversified energy mix.
    - Visualization using PCA demonstrated clear separations between clusters, offering actionable insights into energy consumption trends.

2. **Optimal Cluster Number**:
   - The Elbow and Silhouette methods determined five clusters as the optimal segmentation, balancing computational efficiency and meaningful group distinctions.
3. **Predictive Modeling with Linear Regression**:
   - The regression model highlighted key predictors of energy consumption:
     - **Temperature**: The most significant positive impact, indicating cooling demands in hotter climates.
     - **Occupancy**: Moderate positive correlation due to energy usage driven by building utilization.
     - **Renewable Energy**: A smaller positive impact, underscoring its supplemental role in the energy mix.
   - The model explained nearly 50% of the variance in energy consumption, providing valuable insights but leaving room for incorporating additional factors.
4. **Policy and Strategic Implications**:
   - Clusters with high fossil fuel dependency (e.g., Cluster 0) require infrastructure and policy shifts to accelerate renewable adoption.
   - Advanced nations (e.g., Cluster 2) can serve as models, sharing best practices to guide less-developed regions in their energy transition.

Conclusion:

This project successfully analyzed global energy consumption patterns through a combination of clustering and predictive modeling techniques. By identifying distinct energy profiles, the study provides a roadmap for countries at various stages of the energy transition. The clustering results reveal actionable insights into energy mix compositions, while the regression analysis highlights environmental and operational factors driving energy consumption.

The findings emphasize the importance of tailored energy strategies:

- **Developing nations** should prioritize infrastructure investments and policy incentives to transition from fossil fuels to renewables.
- **Leading nations** can focus on optimizing and scaling renewable technologies while mentoring others in adopting sustainable practices.

While the project delivered meaningful results, future work could focus on refining predictive models to forecast renewable adoption rates, integrating additional variables such as economic and policy indicators. Overall, this analysis demonstrates the power of data-driven approaches to guide global energy transitions and support sustainability goals.