Задачі кластеризації та класифікації

Ознайомитись з різновидами моделей для задач кластеризації та класифікації, а також методами побудови та оцінки цих моделей. Після завершення цієї лабораторної роботи ви зможете:

- Використовувати прості моделі для задач кластеризації та класифікації
- Використовувати перехресну перевірку для оцінки якості моделі
- Обирати оптимальну складність моделі для уникнення перенавчання
- Вдосконалювати моделі за допомогою підбору параметрів

1. Скачайте дані із файлу 'clean_data2.csv' (Data2.csv з виправленими помилками та заповненими пропусками). Виконайте кластеризацію по ВВП на душу населення та щільності населення.
2. Використайте метод ліктя для підбору оптимальної кількості кластерів.
3. Визначіть, який регіон домінує в кожному з кластерів.
4. Побудуйте кілька (3-5) моделей класифікації, що визначають регіон, до якого належить країна, по ознаках 'GDP per capita', 'Population', 'CO2 emission', 'Area'. Оцініть точність класифікації (використайте 20% загального набору в якості тестових даних).
5. Для однієї з моделей виконайте підбір параметра. Обгрунтуйте ваш вибір.

Завдання #1:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split, cross_val_score,
KFold
from sklearn.cluster import KMeans
from scipy.cluster.hierarchy import linkage, dendrogram
from sklearn.metrics import mean_squared_error, r2_score,
classification_report, silhouette_score, confusion_matrix
from sklearn.neighbors import KNeighborsClassifier
from sklearn.svm import SVC
from sklearn import tree
from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import RandomForestClassifier
from sklearn.ensemble import ExtraTreesClassifier
from sklearn.ensemble import AdaBoostClassifier
from sklearn.ensemble import GradientBoostingClassifier
```

Зчитую дані з файлу у датафрейм

```
# Напишіть ваш код нижче та натисніть Shift+Enter для виконання
df = pd.read_csv("clean_data2.csv", encoding='cp1252')
df
```


```
{"summary":"{\n  \"name\": \"df\",\n  \"rows\": 217,\n  \"fields\": [\
n    {\n      \"column\": \"Country Name\",\n      \"properties\": {\n
\"dtype\": \"string\",\n        \"num_unique_values\": 217,\n
\"samples\": [\n          \"United Kingdom\",\n          \"Yemen,
Rep.\",\n          \"Nepal\"\n        ],\n        \"semantic_type\":
\"\",\n        \"description\": \"\"\n      }\n    },\n    {\n
\"column\": \"Region\",\n      \"properties\": {\n        \"dtype\":
\"category\",\n        \"num_unique_values\": 7,\n        \"samples\":
[\n          \"South Asia\",\n          \"Europe & Central Asia\",\n
\"Latin America & Caribbean\"\n        ],\n        \"semantic_type\":
\"\",\n        \"description\": \"\"\n      }\n    },\n    {\n
\"column\": \"GDP per capita\",\n      \"properties\": {\n
\"dtype\": \"number\",\n        \"std\": 17437.20670452666,\n
\"min\": 1.8873365070462016,\n        \"max\": 100738.6842,\n
\"num_unique_values\": 217,\n        \"samples\": [\n
40367.03784,\n          990.334774,\n          729.1222515\
n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"Population\",\n      \"properties\": {\n        \"dtype\":
\"number\",\n        \"std\": 134463782,\n        \"min\": 11097,\n
\"max\": 1378665000,\n        \"num_unique_values\": 217,\n
\"samples\": [\n          65637239,\n          27584213,\n
28982771\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\": \"CO2
emission\",\n      \"properties\": {\n        \"dtype\": \"number\",\n
\"std\": 810928.5931766126,\n        \"min\": 11.001,\n
\"max\": 10291926.88,\n        \"num_unique_values\": 214,\n
\"samples\": [\n          872.746,\n          68422.553,\n
47300.633\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"Area\",\n      \"properties\": {\n        \"dtype\": \"number\",\n
\"std\": 1827830.43486828,\n        \"min\": 2.0,\n        \"max\":
17098250.0,\n        \"num_unique_values\": 213,\n        \"samples\":
[\n          180.0,\n          30.0,\n          338420.0\n        ],\n
\"semantic_type\": \"\",\n        \"description\": \"\"\n      }\
n    },\n    {\n      \"column\": \"Density\",\n      \"properties\":
{\n        \"dtype\": \"number\",\n        \"std\":
2012.959696615876,\n        \"min\": 0.1368887806066512,\n
\"max\": 20203.531353135317,\n        \"num_unique_values\": 217,\n
\"samples\": [\n          269.4357333442798,\n
52.24579616266076,\n          196.92058024188069\n        ],\n
\"semantic_type\": \"\",\n        \"description\": \"\"\n      }\
n    }\n  ]\n}","type":"dataframe","variable_name":"df"}
```


Виділяю параметри для кластеризації

```
features = df[['GDP per capita', 'Density']]
features
```


```
{"summary":"{\n  \"name\": \"features\",\n  \"rows\": 217,\n
\"fields\": [\n    {\n      \"column\": \"GDP per capita\",\n
\"properties\": {\n        \"dtype\": \"number\",\n        \"std\":
17437.20670452666,\n        \"min\": 1.8873365070462016,\n
\"max\": 100738.6842,\n        \"num_unique_values\": 217,\n
\"samples\": [\n          40367.03784,\n          990.334774,\n
729.1222515\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"Density\",\n      \"properties\": {\n        \"dtype\": \"number\",\
n        \"std\": 2012.959696615876,\n        \"min\":
0.1368887806066512,\n        \"max\": 20203.531353135317,\n
\"num_unique_values\": 217,\n        \"samples\": [\n
269.4357333442798,\n          52.24579616266076,\n
196.92058024188069\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    }\n  ]\
n}","type":"dataframe","variable_name":"features"}
```


Будую модель методом k середніх з кількістю кластерів 5

```
kmeans1 = KMeans(
    init='random',
    n_clusters=5,
    n_init=10,
    max_iter=300
)

kmeans1.fit(features)
```

```
KMeans(init='random', n_clusters=5, n_init=10)
```

```
kmeans1.cluster_centers_
```

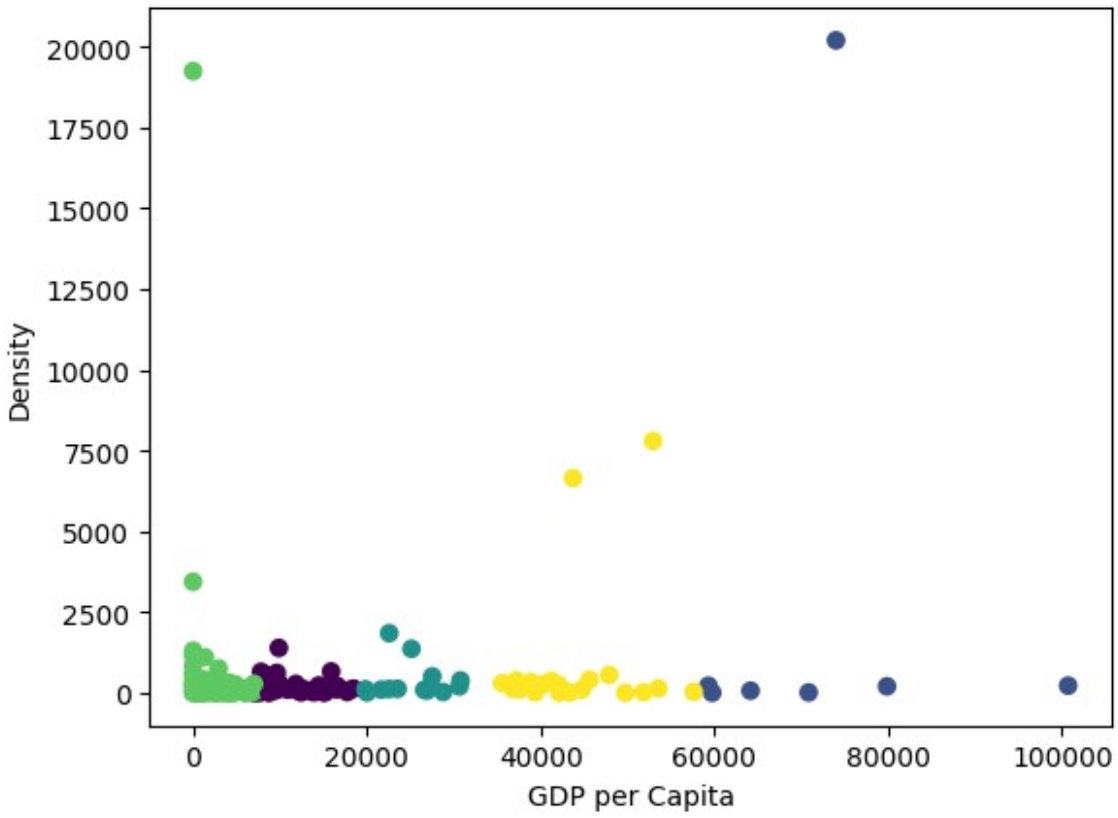
```
array([[12017.47051407,   185.81445875],
       [72682.28457714,  2991.06724725],
       [25288.99915571,   371.88157383],
       [ 2179.26054784,   314.9761539 ],
       [43796.79500476,   858.9640942 ]])
```

```
kmeans1.labels_
```

```
array([3, 3, 3, 0, 4, 3, 0, 0, 3, 3, 4, 4, 3, 2, 2, 3, 0, 3, 4, 3, 3,
3,
       3, 3, 3, 3, 0, 3, 2, 0, 3, 3, 3, 3, 3, 4, 3, 3, 3, 3, 0, 0, 3,
3,
       3, 3, 0, 3, 0, 3, 3, 2, 0, 4, 3, 0, 3, 3, 3, 3, 0, 3, 0, 3, 3,
3,
       4, 4, 3, 0, 3, 3, 4, 3, 3, 0, 3, 0, 4, 3, 3, 3, 3, 3, 3, 4, 0,
```

```
1,
       3, 3, 3, 3, 1, 3, 4, 2, 3, 4, 3, 0, 3, 3, 3, 2, 3, 2, 3, 3, 0,
0,
       3, 3, 3, 3, 0, 1, 1, 3, 3, 3, 0, 0, 3, 2, 3, 3, 0, 0, 3, 3, 3,
3,
       3, 3, 3, 3, 3, 0, 3, 4, 3, 4, 3, 3, 3, 2, 1, 0, 3, 0, 0, 3, 3,
3,
       3, 0, 2, 2, 1, 0, 0, 3, 3, 4, 3, 2, 3, 3, 0, 3, 4, 3, 0, 2, 3,
3,
       3, 3, 2, 3, 0, 0, 3, 3, 3, 3, 3, 4, 1, 3, 3, 3, 3, 3, 3, 3, 0,
3,
       0, 3, 3, 3, 3, 3, 4, 4, 4, 0, 3, 3, 3, 3, 3, 3, 3, 3, 3],
      dtype=int32)
```

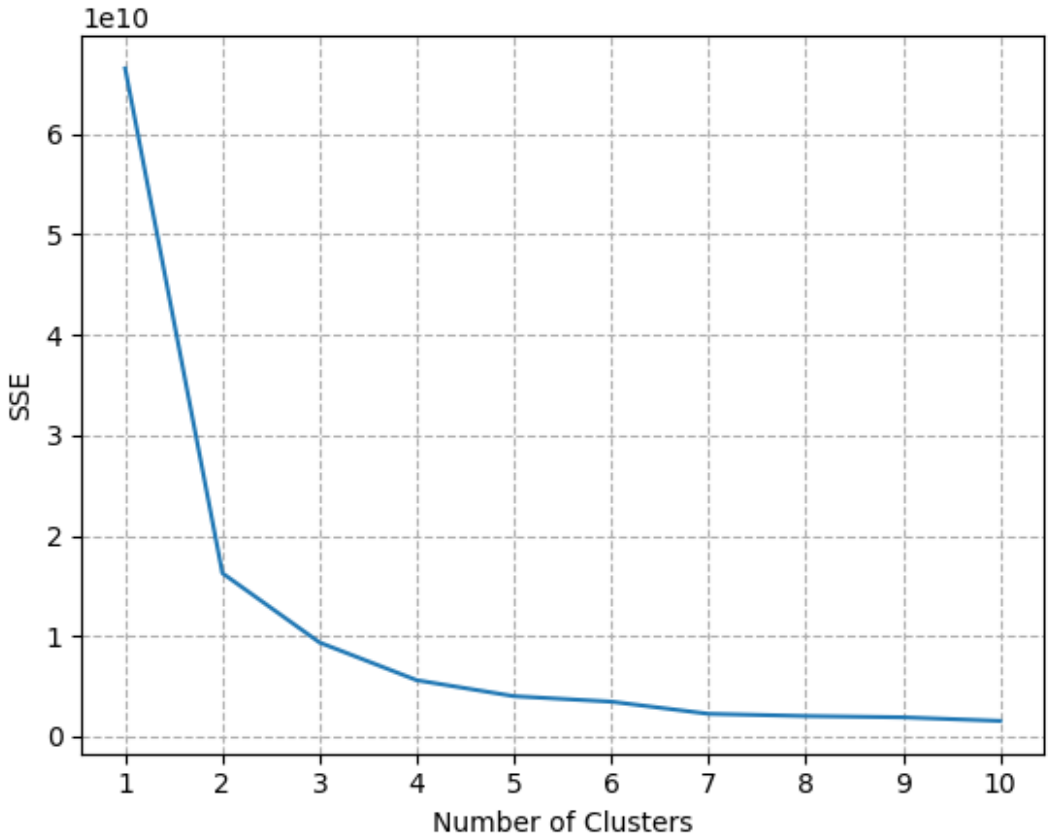
```
plt.xlabel('GDP per Capita')
plt.ylabel('Density')
plt.scatter(df[['GDP per capita']], df[['Density']],
c=kmeans1.labels_)
plt.show()
```



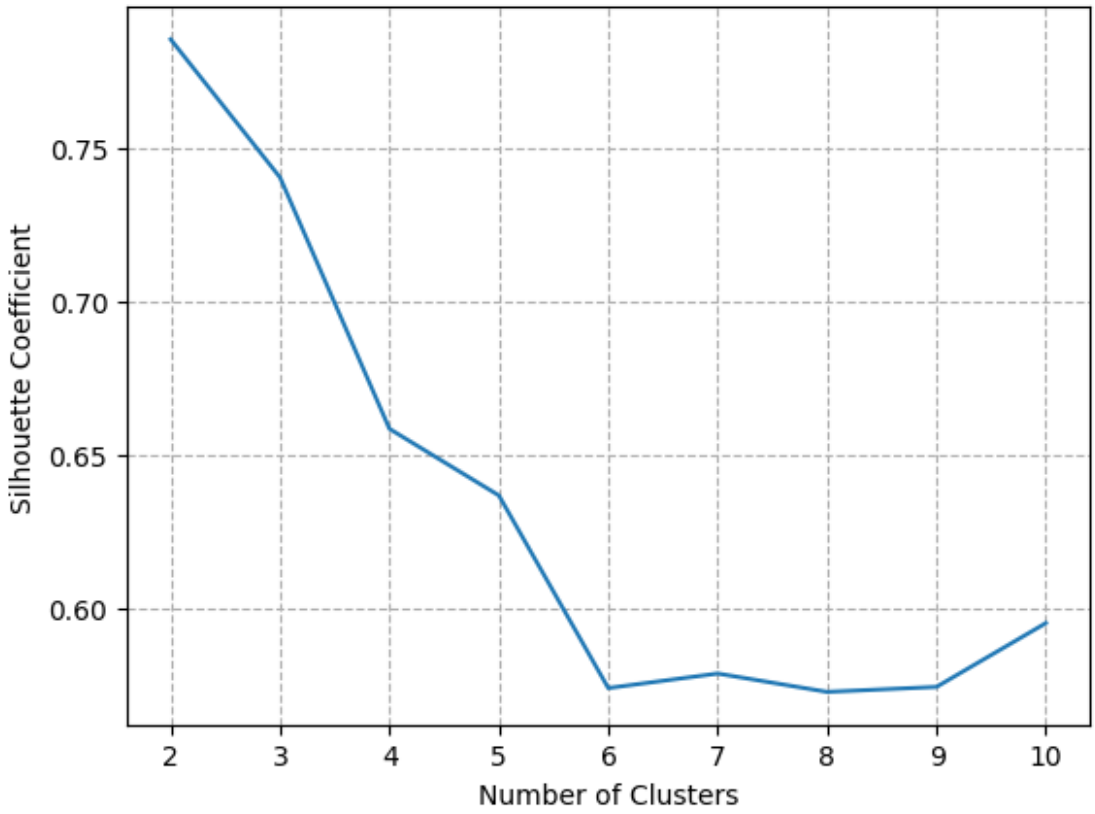
Завдання #2:

Визначаю оптимальну кількість кластерів. Скористаюсь методом "ліктя". Для цього ініціалізую алгоритм k середніх кількістю кластерів від 1 до 10 і для кожної моделі рахую суму квадратів похибок (евклідових відстаней точок кластерів від відповідних центрів):

```
kmeans_kwargs = {
    'init': 'random',
    'n_init': 10,
    'max_iter': 300,
    'random_state': 42,
}

sse = []
max_kernels = 10
for k in range(1, max_kernels + 1):
    kmeans = KMeans(n_clusters=k, **kmeans_kwargs)
    kmeans.fit(features)
    sse.append(kmeans.inertia_)

plt.plot(range(1, max_kernels + 1), sse)
plt.xticks(range(1, max_kernels + 1))
plt.xlabel('Number of Clusters')
plt.ylabel('SSE')
plt.grid(linestyle='--')
plt.show()
```




```
silhouette_coefficients = []

for k in range(2, max_kernels + 1):
    kmeans = KMeans(n_clusters=k, **kmeans_kwargs)
    kmeans.fit(features)
    score = silhouette_score(features, kmeans.labels_)
    silhouette_coefficients.append(score)

plt.plot(range(2, max_kernels + 1), silhouette_coefficients)
plt.xticks(range(2, max_kernels + 1))
plt.xlabel('Number of Clusters')
plt.ylabel('Silhouette Coefficient')
plt.grid(linestyle='--')
plt.show()
```



За даними двох графіків оптимальна кількість клластерів дорівнює 3.

```
kmeans1 = KMeans(
    init='random',
    n_clusters=3,
    n_init=10,
    max_iter=300
)

kmeans1.fit(features)
```

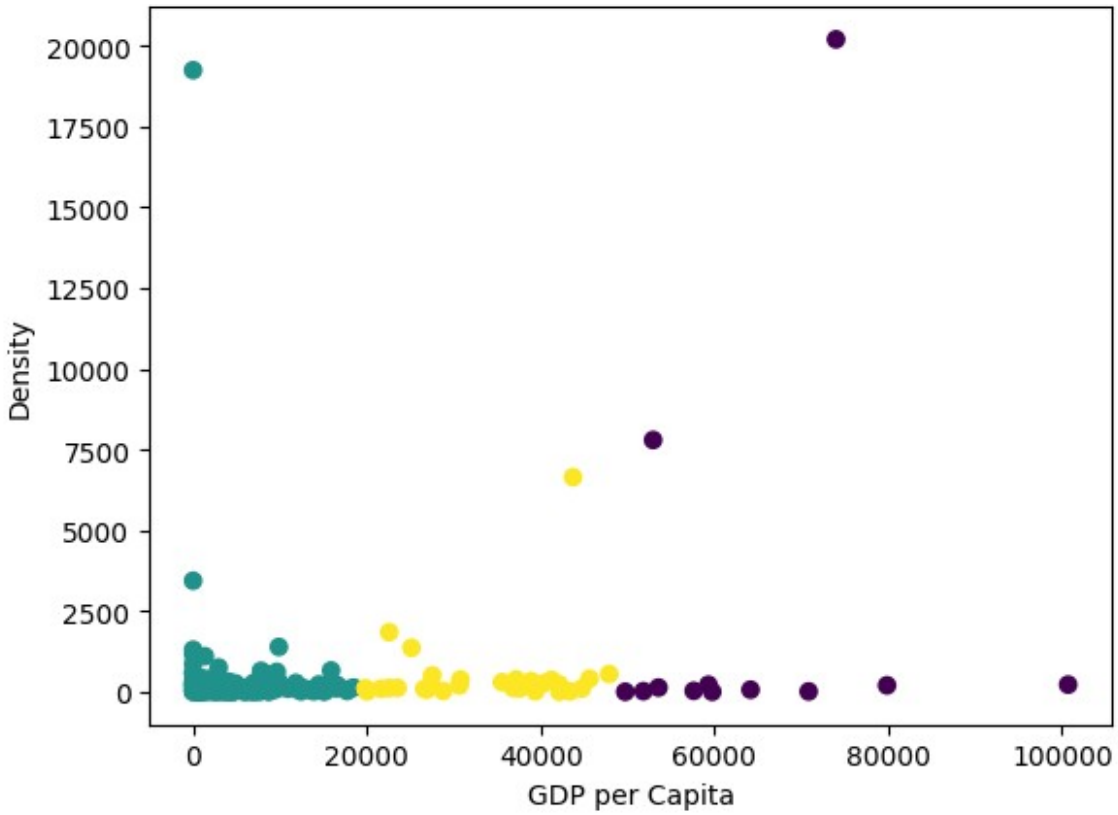
```
KMeans(init='random', n_clusters=3, n_init=10)
```

```
kmeans1.cluster_centers_
kmeans1.labels_
```

```
array([1, 1, 1, 1, 2, 1, 1, 1, 1, 1, 0, 2, 1, 2, 2, 1, 1, 1, 2, 1, 1,
1,
       1, 1, 1, 1, 1, 1, 2, 1, 1, 1, 1, 1, 1, 2, 1, 1, 1, 1, 1, 1, 1,
1,
       1, 1, 1, 1, 1, 1, 1, 2, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1,
       2, 2, 1, 1, 1, 1, 2, 1, 1, 1, 1, 1, 2, 1, 1, 1, 1, 1, 1, 2, 1,
0,
       1, 1, 1, 1, 0, 1, 2, 2, 1, 2, 1, 1, 1, 1, 1, 2, 1, 2, 1, 1, 1,
```

```
1,
       1, 1, 1, 1, 1, 0, 0, 1, 1, 1, 1, 1, 1, 2, 1, 1, 1, 1, 1, 1, 1,
1,
       1, 1, 1, 1, 1, 1, 1, 2, 1, 2, 1, 1, 1, 2, 0, 1, 1, 1, 1, 1, 1,
1,
       1, 1, 2, 2, 0, 1, 1, 1, 1, 2, 1, 2, 1, 1, 1, 1, 0, 1, 1, 2, 1,
1,
       1, 1, 2, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1,
1,
       1, 1, 1, 1, 1, 1, 2, 2, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1],
      dtype=int32)
```

```
plt.xlabel('GDP per Capita')
plt.ylabel('Density')
plt.scatter(df[['GDP per capita']], df[['Density']],
c=kmeans1.labels_)
plt.show()
```



Завдання #3:

Додаю мітки кластерів в датафрейм

```
from sklearn.cluster import KMeans
df['Cluster'] = kmeans1.labels_
df.head()
```


```
{"summary":"{\n  \"name\": \"df\",\n  \"rows\": 217,\n  \"fields\": [\
n    {\n      \"column\": \"Country Name\",\n      \"properties\": {\n
\"dtype\": \"string\",\n        \"num_unique_values\": 217,\n
\"samples\": [\n          \"United Kingdom\",\n          \"Yemen,
Rep.\",\n          \"Nepal\"\n        ],\n        \"semantic_type\":
\"\",\n        \"description\": \"\"\n      }\n    },\n    {\n
\"column\": \"Region\",\n      \"properties\": {\n        \"dtype\":
\"category\",\n        \"num_unique_values\": 7,\n        \"samples\":
[\n          \"South Asia\",\n          \"Europe & Central Asia\",\n
\"Latin America & Caribbean\"\n        ],\n        \"semantic_type\":
\"\",\n        \"description\": \"\"\n      }\n    },\n    {\n
\"column\": \"GDP per capita\",\n      \"properties\": {\n
\"dtype\": \"number\",\n        \"std\": 17437.20670452666,\n
\"min\": 1.8873365070462016,\n        \"max\": 100738.6842,\n
\"num_unique_values\": 217,\n        \"samples\": [\n
40367.03784,\n          990.334774,\n          729.1222515\
n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"Population\",\n      \"properties\": {\n        \"dtype\":
\"number\",\n        \"std\": 134463782,\n        \"min\": 11097,\n
\"max\": 1378665000,\n        \"num_unique_values\": 217,\n
\"samples\": [\n          65637239,\n          27584213,\n
28982771\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\": \"CO2
emission\",\n      \"properties\": {\n        \"dtype\": \"number\",\n
\"std\": 810928.5931766126,\n        \"min\": 11.001,\n
\"max\": 10291926.88,\n        \"num_unique_values\": 214,\n
\"samples\": [\n          872.746,\n          68422.553,\n
47300.633\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"Area\",\n      \"properties\": {\n        \"dtype\": \"number\",\n
\"std\": 1827830.43486828,\n        \"min\": 2.0,\n        \"max\":
17098250.0,\n        \"num_unique_values\": 213,\n        \"samples\":
[\n          180.0,\n          30.0,\n          338420.0\n        ],\n
\"semantic_type\": \"\",\n        \"description\": \"\"\n      }\
n    },\n    {\n      \"column\": \"Density\",\n      \"properties\":
{\n        \"dtype\": \"number\",\n        \"std\":
2012.959696615876,\n        \"min\": 0.1368887806066512,\n
\"max\": 20203.531353135317,\n        \"num_unique_values\": 217,\n
\"samples\": [\n          269.4357333442798,\n
52.24579616266076,\n          196.92058024188069\n        ],\n
\"semantic_type\": \"\",\n        \"description\": \"\"\n      }\
n    },\n    {\n      \"column\": \"Cluster\",\n      \"properties\":
{\n        \"dtype\": \"int32\",\n        \"num_unique_values\": 3,\n
\"samples\": [\n          1,\n          2,\n          0\n        ],\n
```

\"semantic_type\": \"\",\n   \"description\": \"\"\n   }\
n   }\n]\n}","type":"dataframe","variable_name":"df"}

Визначаю домінуючий регіон для кожного кластера

```
cluster_df = df[['Region', 'Cluster']]

cluster_region_counts = cluster_df.groupby(['Cluster',
'Region']).size().reset_index(name='Count')

dominant_regions =
cluster_region_counts.loc[cluster_region_counts.groupby('Cluster')
['Count'].idxmax()]

print(dominant_regions)
```

```
    Cluster                 Region  Count
1         0  Europe & Central Asia      7
10        1     Sub-Saharan Africa     48
12        2  Europe & Central Asia     14
```

Завдання #4:

Обираю потрібні ознаки:

```
all_features=pd.get_dummies(df[['GDP per capita', 'Population', 'CO2
emission', 'Area']])
all_features[['Region']] = df[['Region']]
all_features
```

{"summary":"{\n \"name\": \"all_features\",\n \"rows\": 217,\n
\"fields\": [\n   {\n   \"column\": \"GDP per capita\",\n
\"properties\": {\n   \"dtype\": \"number\",\n   \"std\":
17437.20670452666,\n   \"min\": 1.8873365070462016,\n
\"max\": 100738.6842,\n   \"num_unique_values\": 217,\n
\"samples\": [\n   40367.03784,\n   990.334774,\n
729.1222515\n   ],\n   \"semantic_type\": \"\",\n
\"description\": \"\"\n   }\n   },\n   {\n   \"column\":
\"Population\",\n   \"properties\": {\n   \"dtype\":
\"number\",\n   \"std\": 134463782,\n   \"min\": 11097,\n
\"max\": 1378665000,\n   \"num_unique_values\": 217,\n
\"samples\": [\n   65637239,\n   27584213,\n
28982771\n   ],\n   \"semantic_type\": \"\",\n
\"description\": \"\"\n   }\n   },\n   {\n   \"column\": \"CO2
emission\",\n   \"properties\": {\n   \"dtype\": \"number\",\n
\"std\": 810928.5931766126,\n   \"min\": 11.001,\n
\"max\": 10291926.88,\n   \"num_unique_values\": 214,\n

```
\"samples\": [\n          872.746,\n          68422.553,\n
47300.633\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"Area\",\n      \"properties\": {\n        \"dtype\": \"number\",\n
\"std\": 1827830.43486828,\n        \"min\": 2.0,\n        \"max\":
17098250.0,\n        \"num_unique_values\": 213,\n        \"samples\":
[\n          180.0,\n          30.0,\n          338420.0\n        ],\n
\"semantic_type\": \"\",\n        \"description\": \"\"\n      }\
n    },\n    {\n      \"column\": \"Region\",\n      \"properties\":
{\n        \"dtype\": \"category\",\n        \"num_unique_values\":
7,\n        \"samples\": [\n          \"South Asia\",\n
\"Europe & Central Asia\",\n          \"Latin America & Caribbean\"\n
],\n        \"semantic_type\": \"\",\n        \"description\": \"\"\n
}\n    }\n  ]\n}","type":"dataframe","variable_name":"all_features"}
```


Розділяю датасет на навчальну і тестову вибірки за допомогою функції train_test_split():

```
df_train, df_test = train_test_split(
    all_features,
    test_size=0.2,
    random_state=1
)
df_train.head()
```


```
{"summary":"{\n  \"name\": \"df_train\",\n  \"rows\": 173,\n
\"fields\": [\n    {\n      \"column\": \"GDP per capita\",\n
\"properties\": {\n        \"dtype\": \"number\",\n        \"std\":
17907.462541190038,\n        \"min\": 1.8873365070462016,\n
\"max\": 100738.6842,\n        \"num_unique_values\": 173,\n
\"samples\": [\n          37622.20746,\n          5219.109408,\n
5233.469423\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"Population\",\n      \"properties\": {\n        \"dtype\":
\"number\",\n        \"std\": 149317569,\n        \"min\": 11097,\n
\"max\": 1378665000,\n        \"num_unique_values\": 173,\n
\"samples\": [\n          9269612,\n          80277428,\n
898760\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\": \"CO2
emission\",\n      \"properties\": {\n        \"dtype\": \"number\",\n
\"std\": 894130.9580738026,\n        \"min\": 11.001,\n
\"max\": 10291926.88,\n        \"num_unique_values\": 171,\n
\"samples\": [\n          56372.791,\n          495.045,\n
2467.891\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"Area\",\n      \"properties\": {\n        \"dtype\": \"number\",\n
\"std\": 1419496.693553011,\n        \"min\": 2.0,\n        \"max\":
9831510.0,\n        \"num_unique_values\": 169,\n        \"samples\":
[\n          117600.0,\n          462840.0,\n          410450.0\n
],\n        \"semantic_type\": \"\",\n        \"description\": \"\"\n
```

```
}\n    },\n    {\n      \"column\": \"Region\",\n      \"properties\":
{\n        \"dtype\": \"category\",\n        \"num_unique_values\":
7,\n        \"samples\": [\n          \"Europe & Central Asia\",\n
\"Sub-Saharan Africa\",\n          \"South Asia\"\n        ],\n
\"semantic_type\": \"\",\n        \"description\": \"\"\n      }\
n    }\n  ]\n}","type":"dataframe","variable_name":"df_train"}
```

```
x_train = df_train[['GDP per capita', 'Population', 'CO2 emission',
'Area']]
y_train = df_train[['Region']]

x_test = df_test[['GDP per capita', 'Population', 'CO2 emission',
'Area']]
y_test = df_test[['Region']]
```

Для навчання були обрані наступні методи:

- k-nearest neighbors;
- Decision Tree;
- Random Forest;
- Extra Trees;
- Gradient Boosting.

```
def show_confusion_matrix(matrix, title):
    ax = sns.heatmap(matrix, annot=True, cmap='Greens')
    ax.set_title(title)
    ax.set_xlabel('\nPredicted Values')
    ax.set_ylabel('Actual Values ')
    #ax.xaxis.set_ticklabels(['Negative', 'Positive'])  # тільки для 2
класів
    #ax.yaxis.set_ticklabels(['False', 'True'])         # тільки для 2
класів

    plt.show()
```

k-nearest neighbors

Будую модель

```
KNN_model = KNeighborsClassifier(n_neighbors=20)
KNN_model.fit(x_train, y_train)
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/neighbors/
_classification.py:215: DataConversionWarning: A column-vector y was
passed when a 1d array was expected. Please change the shape of y to
(n_samples,), for example using ravel().
  return self._fit(X, y)
```

```
KNeighborsClassifier(n_neighbors=20)
```

Оцінюю точність

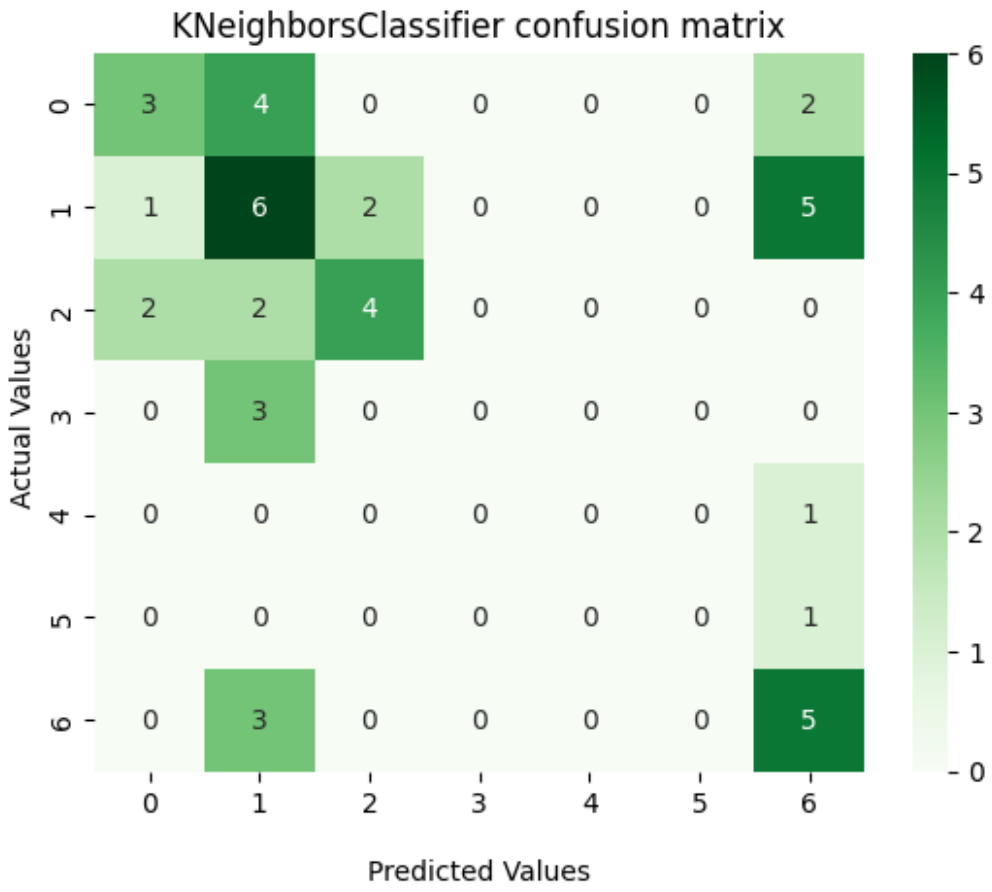
```
print('mean accuracy = ', KNN_model.score(x_test, y_test))
```

```
mean accuracy =  0.4090909090909091
```

```
print(classification_report(y_test, KNN_model.predict(x_test)))
```

```
                            precision    recall  f1-score   support

       East Asia & Pacific       0.50      0.33      0.40         9
     Europe & Central Asia       0.33      0.43      0.38        14
 Latin America & Caribbean       0.67      0.50      0.57         8
Middle East & North Africa       0.00      0.00      0.00         3
             North America       0.00      0.00      0.00         1
                South Asia       0.00      0.00      0.00         1
        Sub-Saharan Africa       0.36      0.62      0.45         8

                  accuracy                           0.41        44
                 macro avg       0.27      0.27      0.26        44
              weighted avg       0.39      0.41      0.39        44


/usr/local/lib/python3.10/dist-packages/sklearn/metrics/
_classification.py:1344: UndefinedMetricWarning: Precision and F-score
are ill-defined and being set to 0.0 in labels with no predicted
samples. Use `zero_division` parameter to control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classificatio
n.py:1344: UndefinedMetricWarning: Precision and F-score are ill-
defined and being set to 0.0 in labels with no predicted samples. Use
`zero_division` parameter to control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classificatio
n.py:1344: UndefinedMetricWarning: Precision and F-score are ill-
defined and being set to 0.0 in labels with no predicted samples. Use
`zero_division` parameter to control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
```

```
show_confusion_matrix(confusion_matrix(y_test,
KNN_model.predict(x_test)), 'KNeighborsClassifier confusion matrix')
```



Decision Tree

Будую модель

```
decision_tree = DecisionTreeClassifier(max_depth=9, random_state=1)
tree_scores = cross_val_score(decision_tree, x_train, y_train, cv=5)
tree_scores
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/
_split.py:700: UserWarning: The least populated class in y has only 2
members, which is less than n_splits=5.
  warnings.warn(

array([0.51428571, 0.42857143, 0.25714286, 0.44117647, 0.38235294])

tree_scores.mean()

0.40470588235294114

decision_tree.fit(x_train, y_train)

DecisionTreeClassifier(max_depth=9, random_state=1)
```

Оцінюю точність

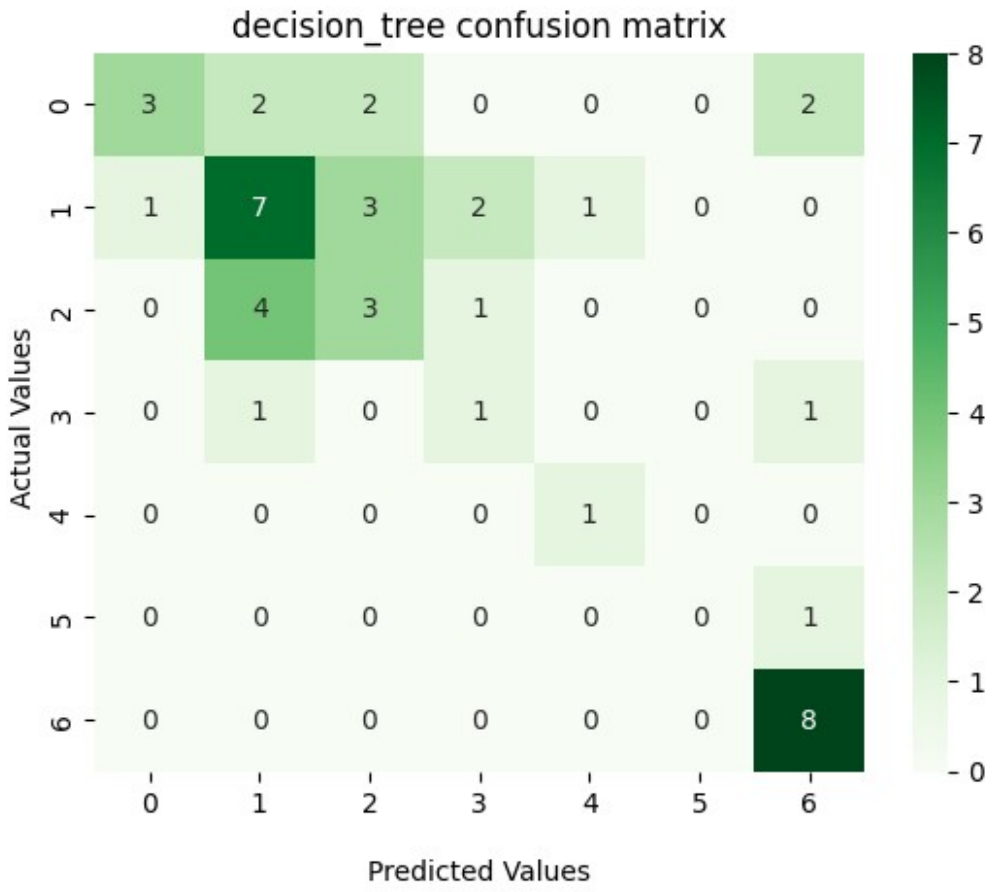
```
decision_tree.score(x_test, y_test)
```

```
0.5227272727272727
```

```
print(classification_report(y_test, decision_tree.predict(x_test)))
```

```
                            precision    recall  f1-score   support

       East Asia & Pacific       0.75      0.33      0.46         9
    Europe & Central Asia       0.50      0.50      0.50        14
 Latin America & Caribbean       0.38      0.38      0.38         8
Middle East & North Africa       0.25      0.33      0.29         3
             North America       0.50      1.00      0.67         1
                South Asia       0.00      0.00      0.00         1
       Sub-Saharan Africa       0.67      1.00      0.80         8

                  accuracy                           0.52        44
                 macro avg       0.43      0.51      0.44        44
              weighted avg       0.53      0.52      0.50        44


/usr/local/lib/python3.10/dist-packages/sklearn/metrics/
_classification.py:1344: UndefinedMetricWarning: Precision and F-score
are ill-defined and being set to 0.0 in labels with no predicted
samples. Use `zero_division` parameter to control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classificatio
n.py:1344: UndefinedMetricWarning: Precision and F-score are ill-
defined and being set to 0.0 in labels with no predicted samples. Use
`zero_division` parameter to control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classificatio
n.py:1344: UndefinedMetricWarning: Precision and F-score are ill-
defined and being set to 0.0 in labels with no predicted samples. Use
`zero_division` parameter to control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
```

```
show_confusion_matrix(confusion_matrix(y_test,
decision_tree.predict(x_test)), 'decision_tree confusion matrix')
```




```
import numpy as np
import matplotlib.pyplot as plt
from sklearn import tree

plt.figure(figsize=(25,20))
_ = tree.plot_tree(decision_tree,
                   feature_names=x_train.columns.tolist(),
                   class_names=np.unique(y_train),
                   filled=True)
```

Random Forest

Будую модель

```
randomforest = RandomForestClassifier(max_depth=2)
random_scores = cross_val_score(randomforest, x_train, y_train, cv=5)
random_scores
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/
_split.py:700: UserWarning: The least populated class in y has only 2
members, which is less than n_splits=5.
  warnings.warn(
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_validation.py:686: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples,), for example using ravel().
  estimator.fit(X_train, y_train, **fit_params)
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_validation.py:686: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples,), for example using ravel().
  estimator.fit(X_train, y_train, **fit_params)
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_validation.py:686: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples,), for example using ravel().
  estimator.fit(X_train, y_train, **fit_params)
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_validation.py:686: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples,), for example using ravel().
  estimator.fit(X_train, y_train, **fit_params)
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_validation.py:686: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples,), for example using ravel().
  estimator.fit(X_train, y_train, **fit_params)
```

```
array([0.45714286, 0.51428571, 0.34285714, 0.5       , 0.41176471])
```

```
random_scores.mean()
```

```
0.4452100840336134
```

```
randomforest.fit(x_train, y_train)
```

```
<ipython-input-34-e1fd2dbde928>:1: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples,), for example using ravel().
  randomforest.fit(x_train, y_train)
```

```
RandomForestClassifier(max_depth=2)
```

Оцінюю точність

```
randomforest.score(x_test, y_test)
```

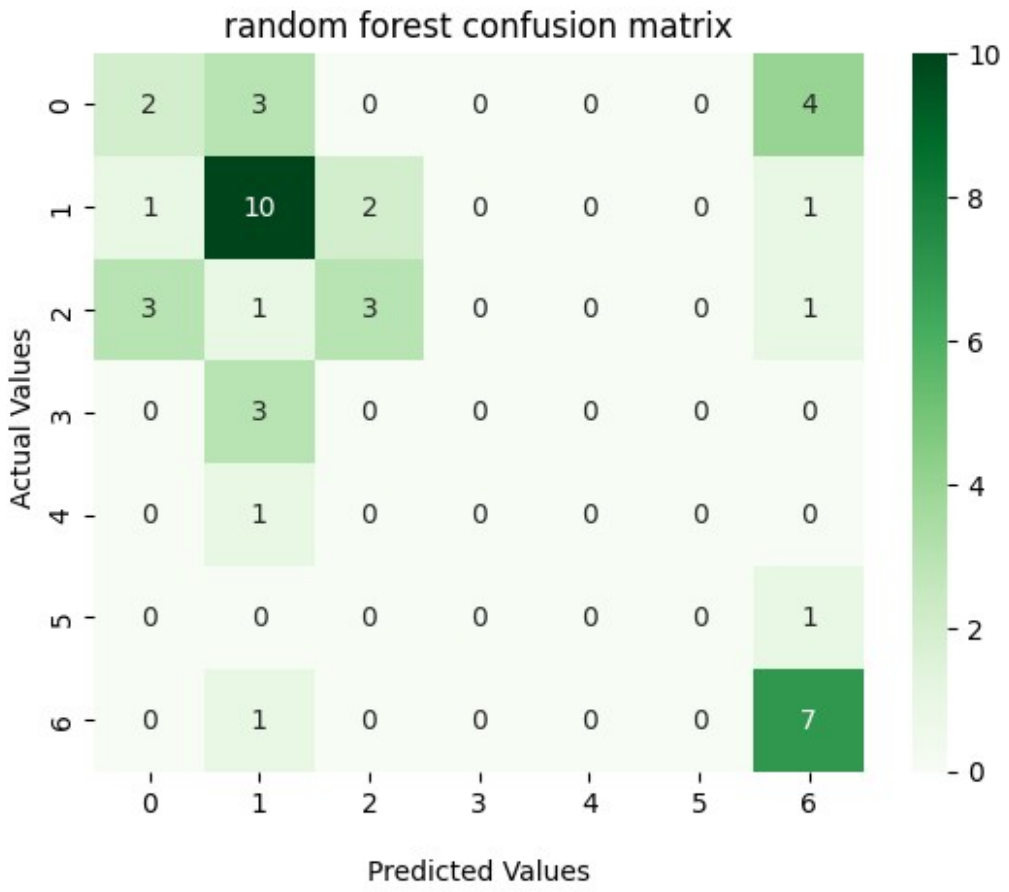
```
0.5
```

```
print(classification_report(y_test, randomforest.predict(x_test)))
```

```
                            precision    recall  f1-score   support
```

```
                East Asia & Pacific       0.33      0.22      0.27         9
             Europe & Central Asia        0.53      0.71      0.61        14
          Latin America & Caribbean       0.60      0.38      0.46         8
Middle East & North Africa                0.00      0.00      0.00         3
                      North America       0.00      0.00      0.00         1
                         South Asia       0.00      0.00      0.00         1
                Sub-Saharan Africa        0.50      0.88      0.64         8

                           accuracy                           0.50        44
                          macro avg       0.28      0.31      0.28        44
                       weighted avg       0.44      0.50      0.45        44


/usr/local/lib/python3.10/dist-packages/sklearn/metrics/
_classification.py:1344: UndefinedMetricWarning: Precision and F-score
are ill-defined and being set to 0.0 in labels with no predicted
samples. Use `zero_division` parameter to control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classificatio
n.py:1344: UndefinedMetricWarning: Precision and F-score are ill-
defined and being set to 0.0 in labels with no predicted samples. Use
`zero_division` parameter to control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classificatio
n.py:1344: UndefinedMetricWarning: Precision and F-score are ill-
defined and being set to 0.0 in labels with no predicted samples. Use
`zero_division` parameter to control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
```

```
show_confusion_matrix(confusion_matrix(y_test,
randomforest.predict(x_test)), 'random forest confusion matrix')
```



Extra Trees

Будую модель

```
extratrees = ExtraTreesClassifier(max_depth=6)
extra_scores = cross_val_score(extratrees, x_train, y_train, cv=5)
extra_scores
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/
_split.py:700: UserWarning: The least populated class in y has only 2
members, which is less than n_splits=5.
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_valid
ation.py:686: DataConversionWarning: A column-vector y was passed when
a 1d array was expected. Please change the shape of y to (n_samples,),
for example using ravel().
  estimator.fit(X_train, y_train, **fit_params)
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_valid
ation.py:686: DataConversionWarning: A column-vector y was passed when
a 1d array was expected. Please change the shape of y to (n_samples,),
for example using ravel().
  estimator.fit(X_train, y_train, **fit_params)
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_valid
```

```
ation.py:686: DataConversionWarning: A column-vector y was passed when
a 1d array was expected. Please change the shape of y to (n_samples,),
for example using ravel().
  estimator.fit(X_train, y_train, **fit_params)
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_valid
ation.py:686: DataConversionWarning: A column-vector y was passed when
a 1d array was expected. Please change the shape of y to (n_samples,),
for example using ravel().
  estimator.fit(X_train, y_train, **fit_params)
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_valid
ation.py:686: DataConversionWarning: A column-vector y was passed when
a 1d array was expected. Please change the shape of y to (n_samples,),
for example using ravel().
  estimator.fit(X_train, y_train, **fit_params)
```

```
array([0.4       , 0.4       , 0.37142857, 0.47058824, 0.35294118])
```

```
extra_scores.mean()
```

```
0.39899159663865547
```

```
extratrees.fit(x_train, y_train)
```

```
<ipython-input-40-f56e48b8fe6a>:1: DataConversionWarning: A column-
vector y was passed when a 1d array was expected. Please change the
shape of y to (n_samples,), for example using ravel().
  extratrees.fit(x_train, y_train)
```

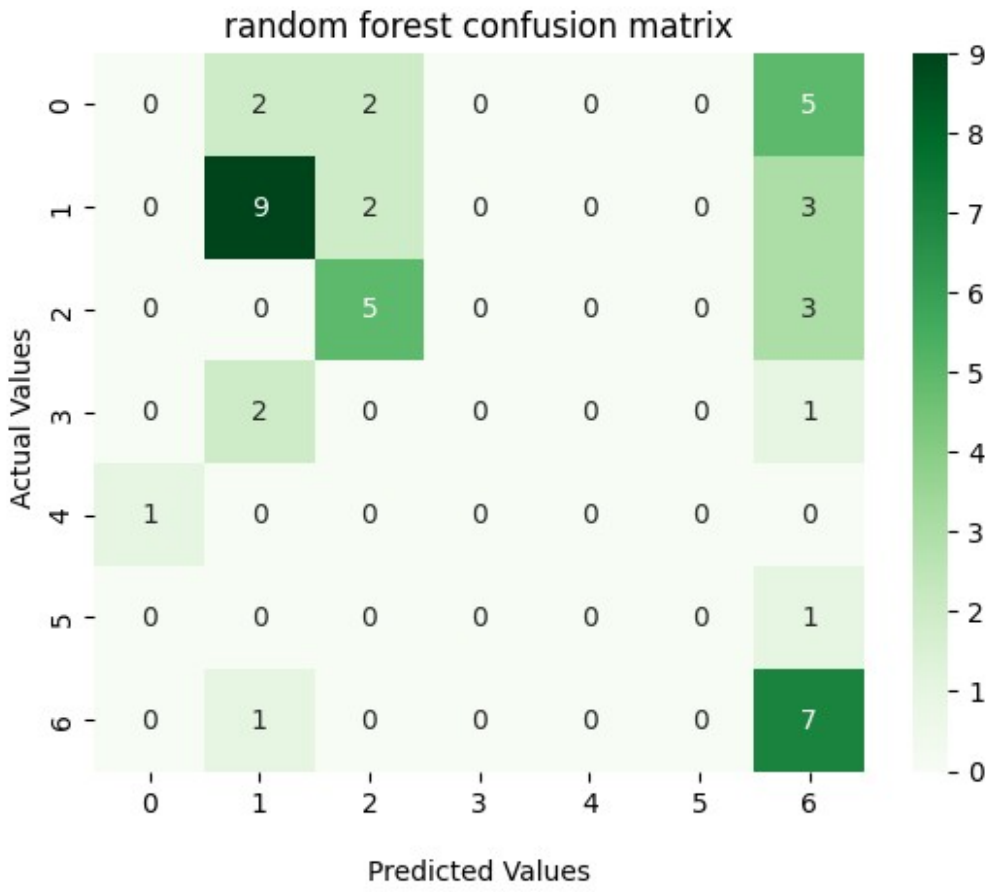
```
ExtraTreesClassifier(max_depth=6)
```

Оцінюю точність

```
extratrees.score(x_test, y_test)
```

```
0.4772727272727273
```

```
show_confusion_matrix(confusion_matrix(y_test,
extratrees.predict(x_test)), 'random forest confusion matrix')
```



Gradient Boosting

Будую модель

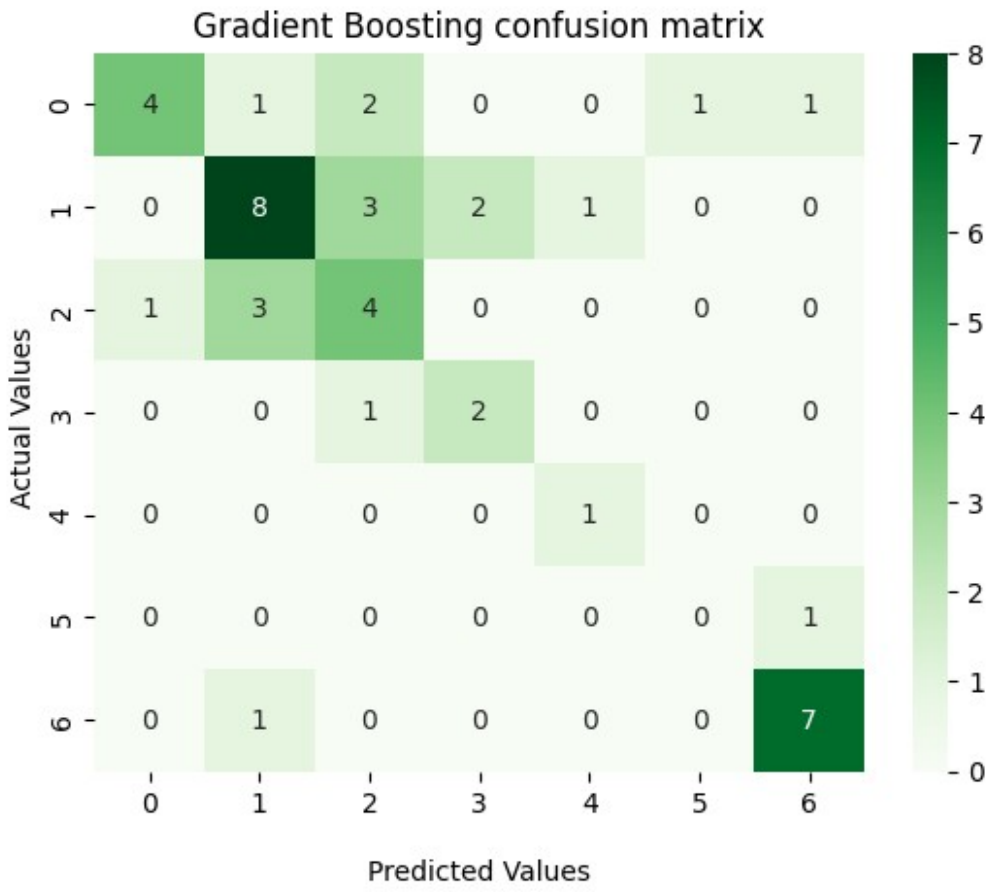
```
gradboost = GradientBoostingClassifier(learning_rate=0.549450)
gradboost_scores = cross_val_score(gradboost, x_train, y_train, cv=5)
gradboost_scores
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/
_split.py:700: UserWarning: The least populated class in y has only 2
members, which is less than n_splits=5.
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
```

```
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)

array([0.34285714, 0.48571429, 0.34285714, 0.5       , 0.47058824])

gradboost_scores.mean()

0.4284033613445378

gradboost.fit(x_train, y_train)

/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)

GradientBoostingClassifier(learning_rate=0.54945)
```

Оцінюю точність

```
gradboost.score(x_test, y_test)

0.5909090909090909

show_confusion_matrix(confusion_matrix(y_test,
gradboost.predict(x_test)), 'Gradient Boosting confusion matrix')
```



Завдання #5:

```
sse = []
for k in range(1, 101):
    gradboost = GradientBoostingClassifier(learning_rate=k/100)
    # gradboost_scores = cross_val_score(gradboost, x_train, y_train,
cv=5)
    gradboost.fit(x_train, y_train)
    sse.append(gradboost.score(x_test, y_test))
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
```

```
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for example using ravel().
```

```
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
```

```
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
```

```
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
```

```
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for example using ravel().
```

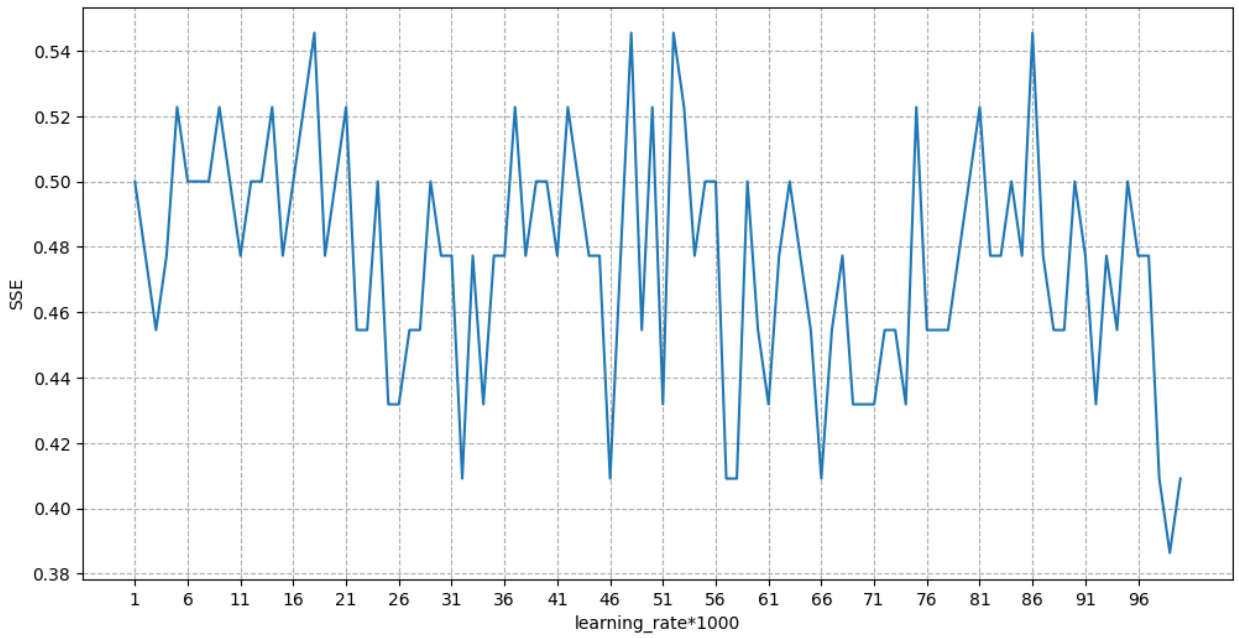
```
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
```

```
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
```

```
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
```

Отримані показники якості візуалізую на графіку:

```
plt.figure(figsize=(12, 6))
plt.plot(range(1, 101), sse)
plt.xticks(range(1, 101, 5))
plt.xlabel('learning_rate*1000')
plt.ylabel('SSE')
plt.grid(linestyle='--')
plt.show()
```



З графіку бачу, що метод ліктя не підходить для визначення параметра learning_rate

Додаткове завдання:

Згідно з методологією вимірювання сталого розвитку країн, сталий розвиток оцінюється за допомогою відповідного індексу у просторі трьох вимірів: економічного (Iec), екологічного (Ie) і соціально-інституціонального (Is). Цей індекс є вектором, норма якого визначає рівень сталого розвитку, а його просторове положення в системі координат (Iec,Ie,Is) характеризує міру «гармонійності» цього розвитку.

```
# Напишіть ваш код нижче та натисніть Shift+Enter для виконання
df = pd.read_csv('Data5.csv', encoding="windows-1251", sep=';',
decimal=',').rename(columns={'Unnamed: 0': 'Country'})
df.head()
```


```
{"summary":"{\n  \"name\": \"df\",\n  \"rows\": 132,\n  \"fields\": [\
n    {\n      \"column\": \"Country\",\n      \"properties\": {\n
\"dtype\": \"string\",\n        \"num_unique_values\": 132,\n
```

```
\"samples\": [\n          \"India\",\n          \"Moldova, Republic
of\",\n          \"Bulgaria\"\n        ],\n        \"semantic_type\":
\"\",\n        \"description\": \"\"\n      }\n    },\n    {\n
\"column\": \"ISO\",\n      \"properties\": {\n        \"dtype\":
\"string\",\n        \"num_unique_values\": 132,\n        \"samples\":
[\n          \"IND\",\n          \"MDA\",\n          \"BGR\"\
n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"UA\",\n      \"properties\": {\n        \"dtype\": \"string\",\n
\"num_unique_values\": 132,\n        \"samples\": [\n          \"\\
u0406\\u043d\\u0434\\u0456\\u044f\",\n          \"\\u041c\\u043e\\
u043b\\u0434\\u043e\\u0432\\u0430, \\u0420\\u0435\\u0441\\u043f\\
u0443\\u0431\\u043b\\u0456\\u043a\\u0430\",\n          \"\\u0411\\
u043e\\u043b\\u0433\\u0430\\u0440\\u0456\\u044f\"\n        ],\n
\"semantic_type\": \"\",\n        \"description\": \"\"\n      }\
n    },\n    {\n      \"column\": \"Cql\",\n      \"properties\": {\n
\"dtype\": \"number\",\n        \"std\": 0.32126377305715037,\n
\"min\": 0.293978537,\n        \"max\": 1.457611163,\n
\"num_unique_values\": 132,\n        \"samples\": [\n
0.640777312,\n          0.905752325,\n          1.135626666\
n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"Ie\",\n      \"properties\": {\n        \"dtype\": \"number\",\n
\"std\": 0.193380402014524,\n        \"min\": 0.133765021,\n
\"max\": 0.822360931,\n        \"num_unique_values\": 129,\n
\"samples\": [\n          0.819964535,\n          0.798270239,\n
0.734293028\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"Iec\",\n      \"properties\": {\n        \"dtype\": \"number\",\n
\"std\": 0.12247925443142166,\n        \"min\": 0.249923267,\n
\"max\": 0.785976304,\n        \"num_unique_values\": 132,\n
\"samples\": [\n          0.450680986,\n          0.404696001,\n
0.555902172\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"Is\",\n      \"properties\": {\n        \"dtype\": \"number\",\n
\"std\": 0.09536835276353604,\n        \"min\": 0.281526245,\n
\"max\": 0.698254396,\n        \"num_unique_values\": 132,\n
\"samples\": [\n          0.4628844,\n          0.531288455,\n
0.554502924\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    }\n  ]\
n}","type":"dataframe","variable_name":"df"}
```


```
kmeans_model = KMeans(n_clusters=4, random_state=1)
df['Cluster_I'] = kmeans_model.fit_predict(df[['Ie','Iec','Is']])
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/
_kmeans.py:870: FutureWarning: The default value of `n_init` will
change from 10 to 'auto' in 1.4. Set the value of `n_init` explicitly
to suppress the warning
  warnings.warn(
```

```
print("Центри кластерів")
print(kmeans_model.cluster_centers_)
```

```
Центри кластерів за Ie, Iec, Is
[[0.26373834 0.39763907 0.40589422]
 [0.67025222 0.51146735 0.52570366]
 [0.50062486 0.43545445 0.47490056]
 [0.76853461 0.66849294 0.63963314]]
```

```
# Напишіть ваш код нижче та натисніть Shift+Enter для виконання
```