(発展) 多次元データのクラスタリング

多次元データのクラスタリング

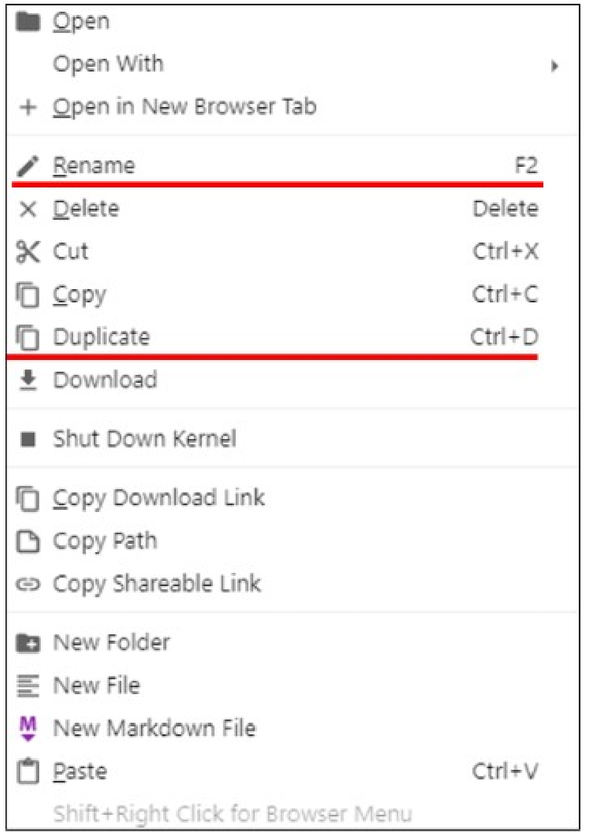
- 2つの指標だけを使った、つまり 2次元データを入力としたk-means法によるクラスタリングをみてきました。
- 実際のデータは、3次元以上になることも多くあります。
- この場合、クラスタリングの様子を2次元の散布図で確認することはできないので、クラスタリング結果の解析として、クラスタごとのデータの分布を把握・比較することになります。
- clustering_country.ipynb をもとに、country_data2019.csv のすべての指標を使った各国のクラスタリングを実行してみましょう。

ノートブックの複製

- プログラムは、clustering_country.ipynb とほとんど同じになります。
- clustering_country.ipynb の作成時と同様に、元のipynbファイルを複製し、それを修正しましょう。

JupyterLab の画面左のファイル・フォルダー一覧の中の clustering_country.ipynb を右クリックして Duplicate を選択すると、clustering_country-Copy1.ipynb が生成します。この clustering_country-Copy1.ipynb を右クリック、Rename を選択し、ファイル名を clustering_country-all.ipynb に変更します。

このclustering_country-all.ipynb をダブルクリックして開きます。



セルの修正

- まず、一番上のノートブックの説明セルを修正しましょう。
  Markdownセルを修正するには、セルをダブルクリックして編集後、
  Ctrl+Enter または Shift+Enter を押します。

```
### Clustering of Country data (use gdpp and imports columns only)
- With elbow method
```

```
### Clustering of Country data (use all columns)
```

セルの修正

- 次に、データフレームdf から、gdpp列とimports列を取り出してデータフレームdfX に代入しているところを修正し、全数値列を用いるようにします。

```
#### Extract data for clustering
```

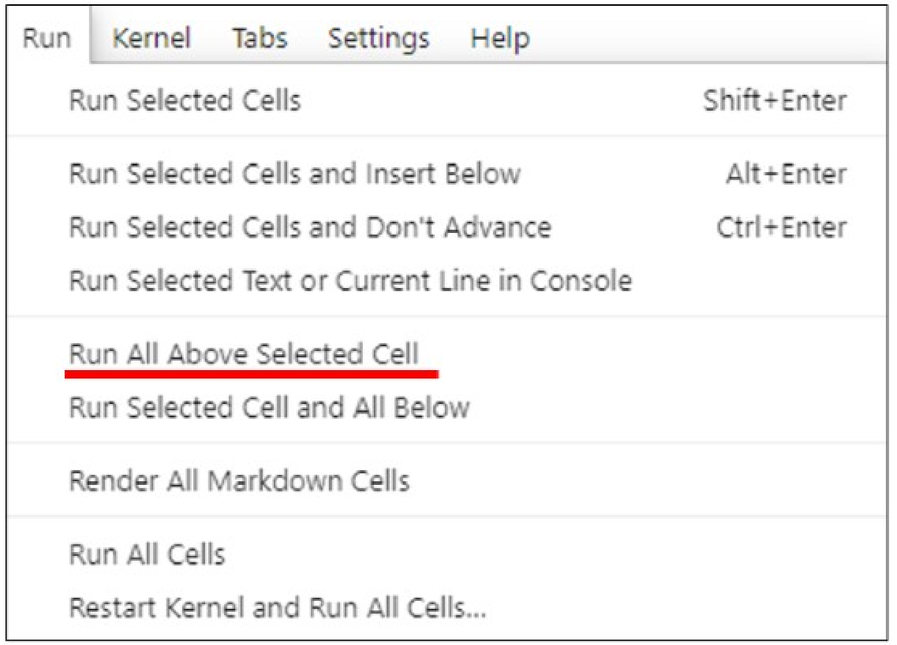
```
dfX = df[['gdpp', 'imports']]
print(dfX.shape)
display(dfX.head())
```

```
#### Extract all numerical data for clustering
```

```
dfX = df.loc[:, 'gdpp':]
print(dfX.shape)
display(dfX.head())
```

ノートブックの部分実行

- セルを修正したら、ノートブックの先頭からこのセルまでのすべてのセルを実行して、結果を確認しましょう。
- 修正したセルをクリックし、Run メニュー > Run All Above Selected Cell を選択すると、ノートブックの先頭から、クリックしたセルの**ひとつ上の**セルまでが自動実行されます。さらに、Shift+Enterを押すと、クリックしたセルも実行できます。



※ クリックしたセルと、それ以降のセルをすべて自動実行するには、Run Selected Cell and All Below を選択します。

gdpp列以降の数値列が、正しくデータフレーム dfX に取り出せていることがわかります。

(151, 9)

	gdpp	gdp_growth	income	imports	exports	health	child_mort	total_fer	life_expec
0	2809.63	-0.62	1700.32	24.94	39.34	2.53	74.2	5.44	61.15
1	42701.44	3.41	39305.78	70.85	96.84	4.28	6.8	1.39	77.97
2	10056.64	-2.03	8160.57	14.52	17.70	9.51	9.1	2.25	76.67
3	4604.65	7.60	4024.14	54.76	41.35	11.34	11.5	1.76	75.09
4	54875.29	2.11	42151.09	21.68	24.17	9.91	3.7	1.66	82.90

散布図関連のセルを削除

- 多次元データの場合は散布図が描画できないので、散布図関連のセルを削除します。

```
#### Scatter plot
```

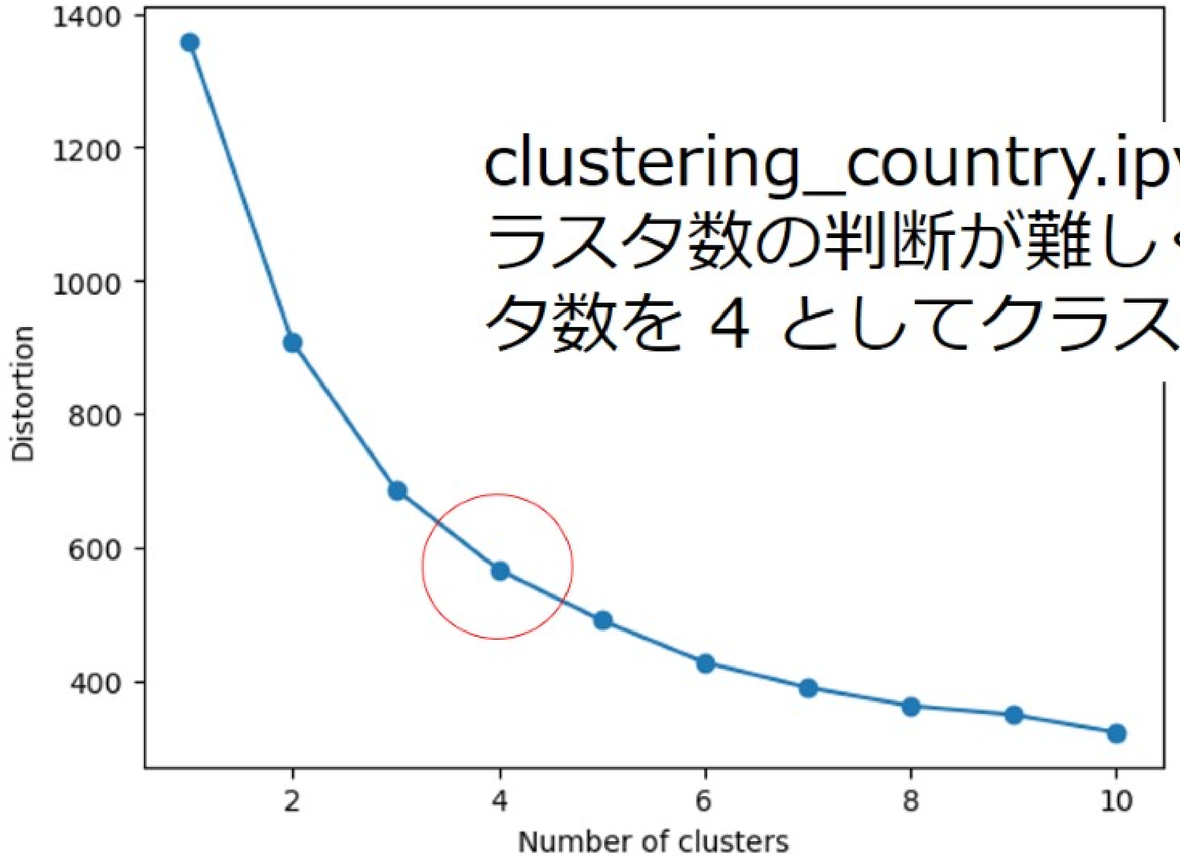
削除

```
plt.scatter(dfX_scaled['gdpp'],
dfX_scaled['imports'], marker='o')
plt.xlabel('gdpp (scaled)')
plt.ylabel('imports (scaled)')
plt.show()
```

削除

Elbow法の結果

Run メニュー > Run All Above Selected Cell と Shift+Enter で、Elbow法を実行しましょう。



clustering_country.ipynb と比べて、さらに妥当なクラスタ数の判断が難しくなっていますが、今回もクラスタ数を 4 としてクラスタリングを実行してみましょう。

クラスタリング結果の解析

「Check the number of cluster members」のセルの出力

0	83
1	42
3	22
2	4

この結果、およびその次の「Check the members of each cluster」セルの出力から、クラスタ番号は付け変わっているが、国の分かれ方はclustering_country.ipynb とよく似ていることがわかります。

clustering_country.ipynbの結果

0	84
2	40
3	24
1	3

clustering_country-all.ipynbの結果

0	83
1	42
3	22
2	4

クラスタリングは、データの分かれ方が本質的。
クラスタ番号そのものには意味はない。

以降の5つのセルは使わないので削除

```
#### Scatter plot (not scaled)
```

削除

```
plt.scatter(df['gdpp'], df['imports'],
                marker='o', c=df['cluster_no'])
plt.colorbar()
plt.xlabel('gdpp')
plt.ylabel('imports')
plt.show()
```

削除

⋮

(以降削除)

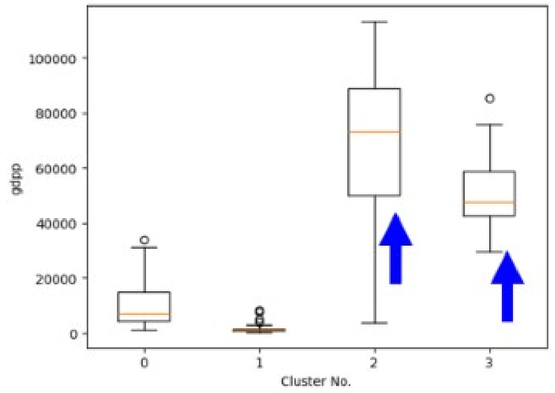
箱ひげ図による可視化

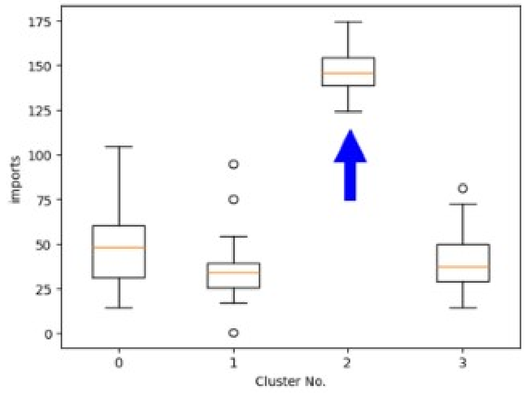
- 各クラスタの特徴を、箱ひげ図で確認してみましょう。
- 各指標のクラスタごとの分布の様子をクラスタ間で比較します。

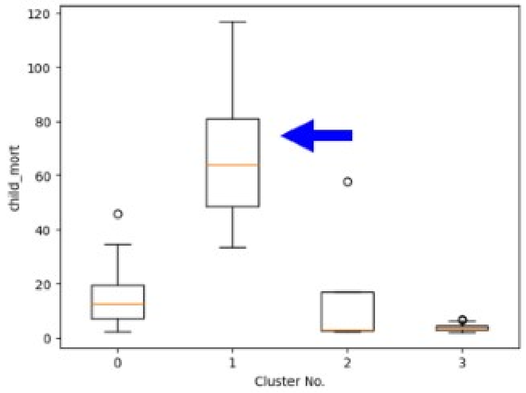
```
#### Box plots (not scaled)
```

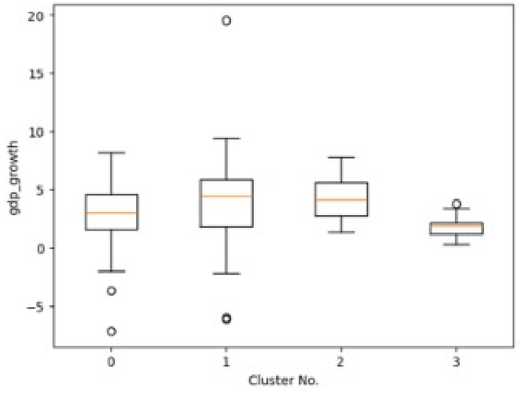
```
cols = list(dfX.columns) クラスタリングに用いた列名(指標名) をリストで取得
for  c  in cols:  各指標(列) を変数 c に次々と入れて回すforループ
    dat = []
    for  i  in range(n_cls):  クラスタ番号を変数 i に次々と入れて回すforループ
        df_cls = df[ df['cluster_no']==i ]  クラスタ i のメンバーだけを df_cls に取り出し
        dat.append(df_cls[c])  列 c をdf_cls から取り出してリストdat にappend
    plt.boxplot(dat, labels=range(n_cls))
    plt.xlabel('Cluster No.')
    plt.ylabel(c)  y 軸のラベルは指標(列)名 c
    plt.show()
```

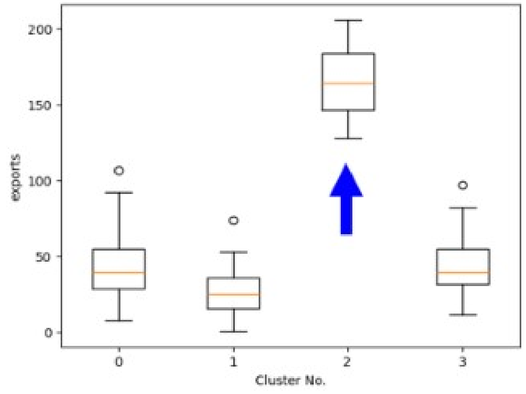
各クラスタの列 c のデータが得られたので、boxplot描画。x軸のラベルはクラスタ番号(rangeで0からn_cls-1までの連番を生成している)。

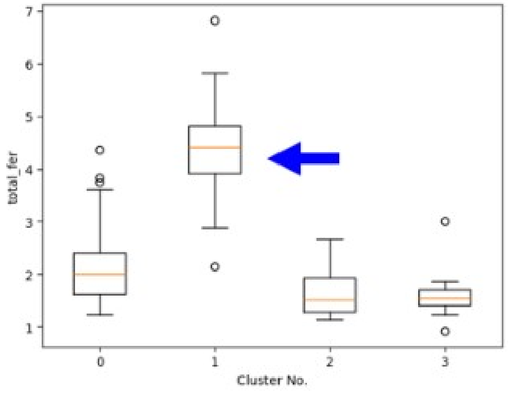


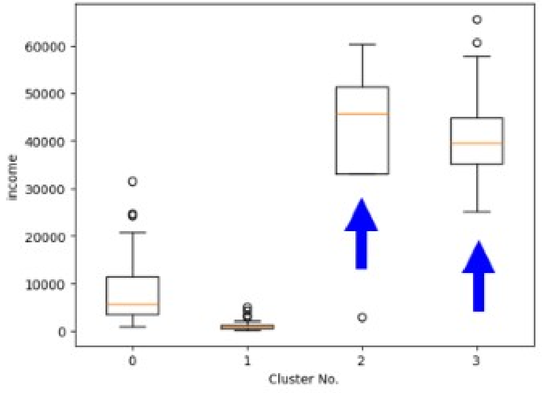


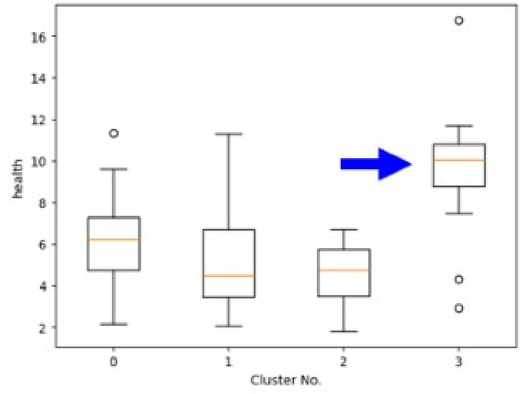


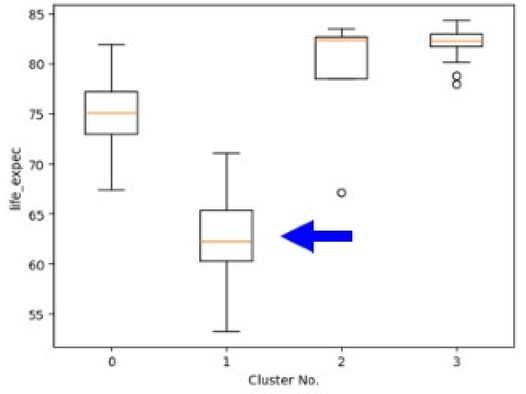












各指標とクラスタの関係

- gdpp (国民1人あたりGDP) が、Cluster 1 < 0 < 3 < 2
- gdpp_growth (GDP成長率) はクラスタ間で大差なし
- life_expect (平均寿命) が、Cluster 1 < 0 < 2 ~ 3
  - ※ 「~」は数学で「大体同じ」という意味
- その他の各クラスタの特徴
  - Cluster 0
    - income (国民所得): 小
  - Cluster 1
    - income (国民所得): 小, child_mort (5歳児未満死亡率): 大, total_fer (出生率): 大
  - Cluster 2
    - income (国民所得): 大, imports (輸入の対GDP割合): 大, exports (輸出の対GDP割合): 大
  - Cluster 3
    - income (国民所得): 大, health (医療費の対GDP割合): 大