課題 1

cs3-06-assign1

データ market.csv, ID_data.csv はそれぞれ、ある小売店の30日間の売り上げおよび顧客属性のデータである。これらを用いて、以下の操作を行うスクリプトをJupyterで作成せよ(ノートブック名は cs3-06-assign1.ipynb/html とせよ)

1. 必要なライブラリをimport。
2. market.csv, ID_data.csv のデータをデータフレーム df_market, df_id にそれぞれ読み込み、行数と列数、各列のデータ型と欠損値でないデータの数、先頭5行と末尾5行を表示して確認。
3. df_market と df_id を、「顧客ID」列をキーとして左外結合で結合し、df に代入。dfの先頭5行を表示して確認。
4. dfから「個数」列と「税抜価格」列のみを抜き出したデータフレーム dfX を作成する。行数と列数、先頭5行を表示して確認。
5. dfXの各列を平均0、母標準偏差(偏差二乗和をデータ数Nで割った分散の平方根)1に標準化し、変数 X_scaled に代入。
6. 手順5 で得た X_scaled の平均と母標準偏差を表示。

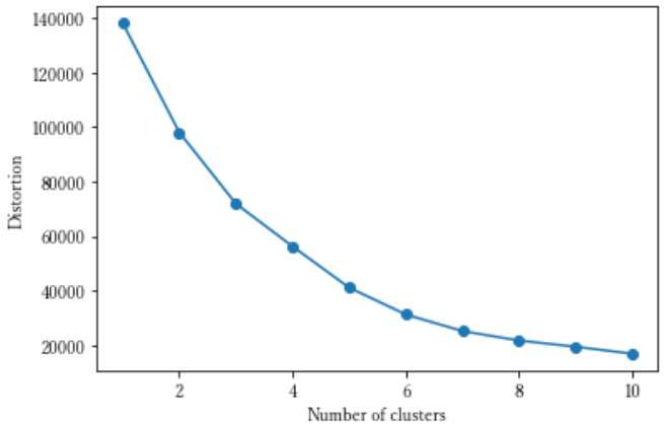
7. 手順5 で得た X_scaled のデータ型と、行数・列数を表示。
8. X_scaled に dfX と同じ列ラベルを付与したデータフレーム dfX_scaled を作成し、データ型を確認、さらに先頭5行を表示。
9. dfX_scaled に対して、KMeans法 (n_init=10とする) によるElbow法を、最大クラスタ数10で実施。クラスタ数を横軸、Inertiaを縦軸とするグラフを描画。
10. dfX_scaled に対してクラスタ数4で KMeans法によるクラスタリングを実行。このとき、n_init=10, random_state=5 とせよ。クラスタリング結果を変数 cls に代入、表示して確認。先頭のデータが割り当てられたクラスタ番号を答えよ。
11. 元のデータフレームdfに、新たな列「cluster_no」を追加し、各データが属するクラスタの番号を格納する。先頭5行を表示して確認。
12. 「cluster_no」列の各値の出現数(各クラスタのメンバー数) を表示。
13. dfの「個数」、「税抜価格」をそれぞれ横軸、縦軸として散布図を描画。クラスタごとに色をつけて区別できるようにせよ。

cs3-06-assign1：出力イメージ

手順3

	レシートNo	日	時間	顧客ID	税抜価格	税抜単価	個数	大カテゴリ番号	大カテゴリ名	中カテゴリ番号	中カテゴリ名	小カテゴリ番号	小カテゴリ名	性別	年代
0	1	1	9	1518	50	10	5	11	農産	1113	野菜	111327	じゃが芋	2	60
1	1	1	9	1518	50	10	5	11	農産	1113	野菜	111363	玉葱	2	60
2	1	1	9	1518	90	90	1	11	農産	1113	野菜	111361	レタス	2	60
3	1	1	9	1518	185	185	1	11	農産	1113	野菜	111339	トマト	2	60
4	2	1	9	1532	85	85	1	11	農産	1113	野菜	111318	キャベツ	1	40

手順9

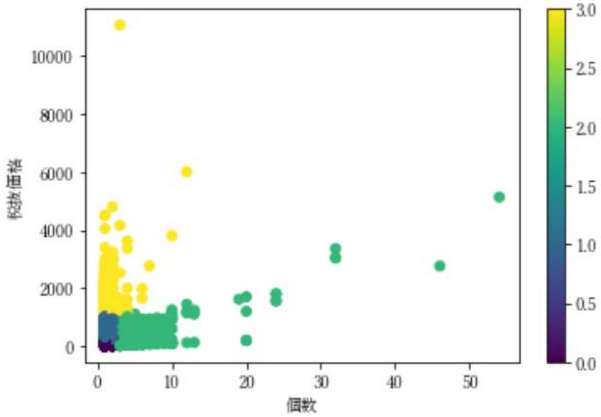


cs3-06-assign1：出力イメージ

手順11

	レシートNo	日	時間	顧客ID	税抜価格	税抜単価	個数	大カテゴリ番号	大カテゴリ名	中カテゴリ番号	中カテゴリ名	小カテゴリ番号	小カテゴリ名	性別	年代	cluster_no
0	1	1	9	1518	50	10	5	11	農産	1113	野菜	111327	じゃが芋	2	60	1
1	1	1	9	1518	50	10	5	11	農産	1113	野菜	111363	玉葱	2	60	1
2	1	1	9	1518	90	90	1	11	農産	1113	野菜	111361	レタス	2	60	0
3	1	1	9	1518	185	185	1	11	農産	1113	野菜	111339	トマト	2	60	0
4	2	1	9	1532	85	85	1	11	農産	1113	野菜	111318	キャベツ	1	40	0

手順13



課題1 正解例

cs3-06-assign1.ipynb/html

1

```
import os
os.environ['OMP_NUM_THREADS'] = '1'

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.cluster import KMeans
from sklearn.preprocessing import scale
```

```
# To show all rows/columns
pd.options.display.max_columns = 999
pd.options.display.max_rows = 999
```

課題1 正解例

cs3-06-assign1.ipynb/html

2

```
csv_market = 'market.csv'
df_market = pd.read_csv(csv_market, sep=',', skiprows=0,
                                    header=0, encoding='shift-jis')
print(df_market.shape)
print(df_market.info())
display(df_market.head())
display(df_market.tail())
```

```
csv_id = 'ID_data.csv'
df_id = pd.read_csv(csv_id, sep=',', skiprows=0,
                            header=0, encoding='shift-jis')
print(df_id.shape)
print(df_id.info())
display(df_id.head())
display(df_id.tail())
```

課題1 正解例

cs3-06-assign1.ipynb/html

3
```
df = pd.merge(df_market, df_id, how='left', on='顧客ID')
display(df.head())
```

4
```
dfX = df[['個数', '税抜価格']]
print(dfX.shape)
display(dfX.head())
```

5
```
X_scaled = scale(dfX)
```

6
```
print(X_scaled.mean(axis=0))
print(X_scaled.std(ddof=0, axis=0))
```

7
```
print(type(X_scaled))
print(X_scaled.shape)
```

8
```
dfX_scaled = pd.DataFrame(X_scaled, columns=dfX.columns)
print(type(dfX_scaled))
display(dfX_scaled.head())
```

課題1 正解例

cs3-06-assign1.ipynb/html

9

```
distortions = []
for i in range(1, 11):
    km = KMeans(n_clusters=i, n_init=10)
    km.fit(dfX_scaled)
    distortions.append(km.inertia_)
plt.plot(range(1, 11), distortions, marker='o')
plt.xlabel('Number of clusters')
plt.ylabel('Distortion')
plt.show()
```

10

```
n_cls = 4
km = KMeans(n_clusters=n_cls, n_init=10, random_state=5)
cls = km.fit_predict(dfX_scaled)
print(cls)
```

```
[2 2 0 ... 0 0 1]
```

答え: クラスタ番号**2**

課題1 正解例

cs3-06-assign1.ipynb/html

11
```
df['cluster_no'] = cls
display(df.head())
```

12
```
print(df['cluster_no'].value_counts())
```

13
```
plt.rcParams['font.family'] = 'Yu Mincho'
plt.scatter(df['個数'], df['税抜価格'],
                marker='o', c=df['cluster_no'])
plt.colorbar()
plt.xlabel('個数')
plt.ylabel('税抜価格')
plt.show()
```

(発展) 課題 2

(Adv) cs3-06-assign2

データ winequality-red.csv は、ワインの品質に関するデータである。これを用いて、以下の操作を行うスクリプトをJupyterで作成せよ(ノートブック名は cs3-06-assign2.ipynb/html とせよ)

1. 必要なライブラリをimportする。
2. winequality-red.csv のデータをデータフレーム df に読み込み、行数と列数、各列のデータ型と欠損値でないデータの数、先頭5行を表示して確認。
3. df の各列の統計量を表示 (describe()を用いる)。
4. df の各列を平均0, 母標準偏差(偏差二乗和をデータ数Nで割った分散の平方根)1に標準化し、変数 X_scaled に代入。
5. 手順5 で得た X_scaled の平均と母標準偏差を表示。
6. X_scaledに、df と同じ列ラベルを付与したデータフレーム df_scaled を作成し、先頭5行を表示。

(Adv) cs3-06-assign2

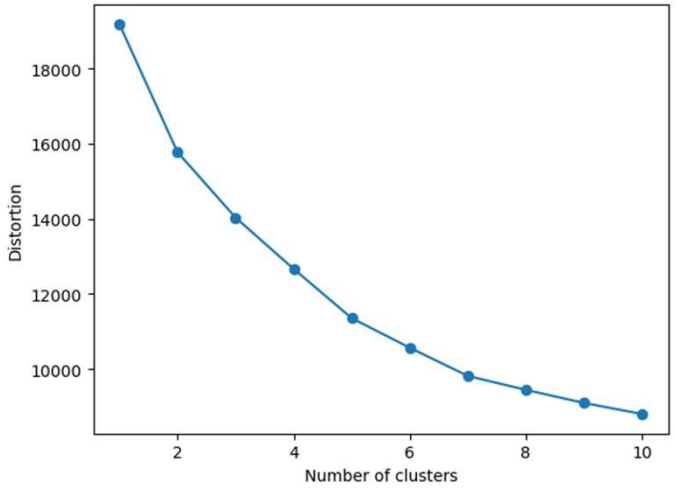
7. df_scaledに対して、KMeans法 (n_init=10とする) によるElbow法を、最大クラスタ数10で実施。クラスタ数を横軸、Inertiaを縦軸とするグラフを描画。
8. データフレーム df_scaled に対してクラスタ数4で KMeans法によるクラスタリングを実行。このとき、n_init=10, random_state=5とせよ。
9. 元のデータフレームdfに、新たな列 cluster_no を追加し、各データが属するクラスタの番号を格納する。先頭5行を表示して確認。
10. 「cluster_no」列の各値の出現数(各クラスタのメンバー数) を表示。
11. 「cluster_no」列以外の列について、クラスタごとの分布を箱ひげ図で確認。クラスタ番号3 が他のクラスタと区別されるのにもっとも大きく寄与していると思われる列を説明せよ。

cs3-06-assign2：出力イメージ

手順3

	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density
count	1599.000000	1599.000000	1599.000000	1599.000000	1599.000000	1599.000000	1599.000000	1599.000000
mean	8.319637	0.527821	0.270976	2.538806	0.087467	15.874922	46.467792	0.996747
std	1.741096	0.179060	0.194801	1.409928	0.047065	10.460157	32.895324	0.001887
min	4.600000	0.120000	0.000000	0.900000	0.012000	1.000000	6.000000	0.990070
25%	7.100000	0.390000	0.090000	1.900000	0.070000	7.000000	22.000000	0.995600
50%	7.900000	0.520000	0.260000	2.200000	0.079000	14.000000	38.000000	0.996750
75%	9.200000	0.640000	0.420000	2.600000	0.090000	21.000000	62.000000	0.997835
max	15.900000	1.580000	1.000000	15.500000	0.611000	72.000000	289.000000	1.003690

手順7

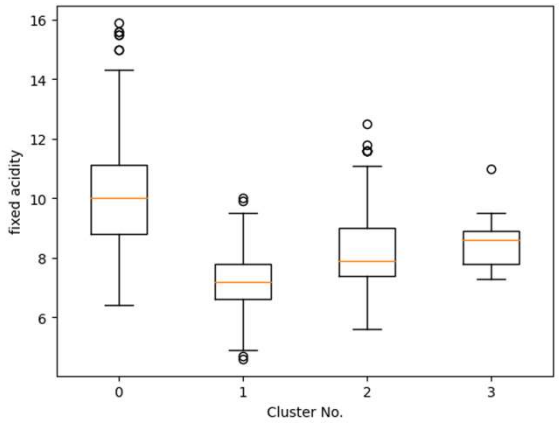


cs3-06-assign2：出力イメージ

手順9

	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sulphates	alcohol	quality	cluster_no
0	7.4	0.70	0.00	1.9	0.076	11.0	34.0	0.9978	3.51	0.56	9.4	5	1
1	7.8	0.88	0.00	2.6	0.098	25.0	67.0	0.9968	3.20	0.68	9.8	5	2
2	7.8	0.76	0.04	2.3	0.092	15.0	54.0	0.9970	3.26	0.65	9.8	5	1
3	11.2	0.28	0.56	1.9	0.075	17.0	60.0	0.9980	3.16	0.58	9.8	6	0
4	7.4	0.70	0.00	1.9	0.076	11.0	34.0	0.9978	3.51	0.56	9.4	5	1

手順11



課題2 正解例

cs3-06-assign2.ipynb/html

1

```
import os
os.environ['OMP_NUM_THREADS'] = '1'

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.cluster import KMeans
from sklearn.preprocessing import scale
```

```
# To show all rows/columns
pd.options.display.max_columns = 999
pd.options.display.max_rows = 999
```

課題2 正解例

cs3-06-assign2.ipynb/html

2
```
csv_in = 'winequality-red.csv'
df = pd.read_csv(csv_in, sep=';', skiprows=0, header=0)
print(df.shape)
print(df.info())
display(df.head())
```

3
```
display(df.describe())
```

4
```
X_scaled = scale(dfX)
```

5
```
print(X_scaled.mean(axis=0))
print(X_scaled.std(ddof=0, axis=0))
```

6
```
df_scaled = pd.DataFrame(X_scaled, columns=df.columns)
display(df_scaled.head())
```

課題2 正解例

cs3-06-assign2.ipynb/html

7

```
distortions = []
for i in range(1, 11):
    km = KMeans(n_clusters=i, n_init=10)
    km.fit(df_scaled)
    distortions.append(km.inertia_)
plt.plot(range(1, 11), distortions, marker='o')
plt.xlabel('Number of clusters')
plt.ylabel('Distortion')
plt.show()
```

8

```
n_cls = 4
km = KMeans(n_clusters=n_cls, n_init=10, random_state=5)
cls = km.fit_predict(df_scaled)
print(cls)
```

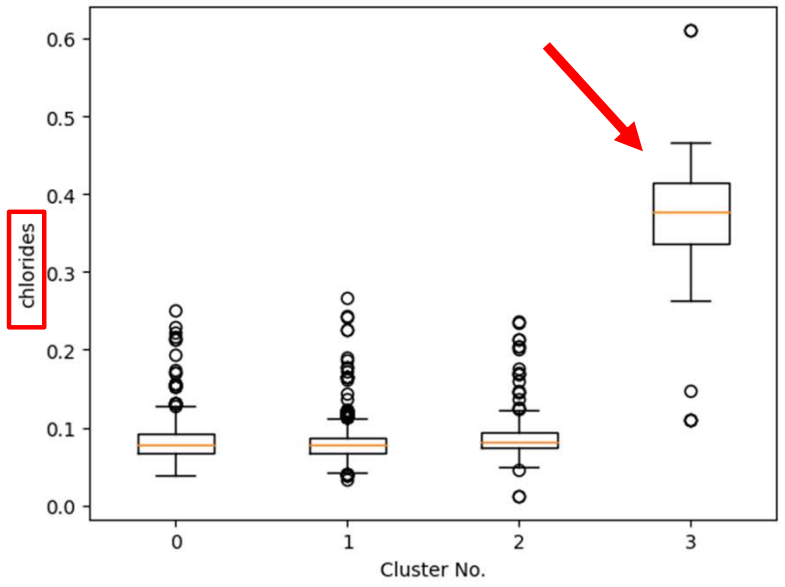
課題2 正解例

cs3-06-assign2.ipynb/html

9
```
df['cluster_no'] = cls
display(df.head())
```

10
```
print(df['cluster_no'].value_counts())
```

11
```
cols = list(df.columns)
for c in cols:
    if c == 'cluster_no': continue
    dat = []
    for i in range(n_cls):
        df_cls = df[ df['cluster_no']==i ]
        dat.append(df_cls[c])
    plt.boxplot(dat, labels=range(n_cls))
    plt.xlabel('Cluster No.')
    plt.ylabel(c)
    plt.show()
```



答え: chlorides