第 3 章: 測定
(3.7. クラスター化)

今井耕介 著

『社会科学のためのデータ分析入門 (QSS)』

2026-03-09

3.7 クラスター化 (Clustering)

クラスター分析の目的

- **クラスター分析**: 多数のデータの中から、互いに似た特徴を持つグループ（クラスター）を自動的に見つけ出す手法。
- **教師なし学習**: 正解（ラベル）があらかじめ与えられていない状態で、データの構造を探索する。
- 社会科学における例：
  - 投票行動が似ている議員のグループ化。
  - 似たような意識を持つ回答者のセグメンテーション。
- 今回は、代表的な手法である **k-means 法**（k 平均法）を学びます。

3.7.1 Rでの行列操作

行列 (Matrix) の作成

- 数値計算を効率的に行うために、同じ型のデータを並べた「行列」を使用します。

```
# 1. 1 から 12 までの数値で 3 行 4 列の行列を作成
# byrow = TRUE で行方向に値を埋める
x <- matrix(1:12, nrow = 3, ncol = 4, byrow = TRUE)

# 2. 行名と列名をつける
rownames(x) <- c("a", "b", "c")
colnames(x) <- c("d", "e", "f", "g")

# 3. 行列の表示と次元の確認
x
```

```
##   d  e  f  g
## a 1  2  3  4
## b 5  6  7  8
## c 9 10 11 12
```

```
dim(x)
```

```
## [1] 3 4
```

行・列ごとの計算 (apply 関数)

- 行列の各行や各列に対して、一括で関数を適用します。

```
# 1. 各列の合計を計算 (2 は「列」を意味する)
apply(x, 2, sum)
```

```
##  d  e  f  g
## 15 18 21 24
```

```
# 2. 各行の平均を計算 (1 は「行」を意味する)
apply(x, 1, mean)
```

```
##    a    b    c
##  2.5  6.5 10.5
```

```
# 3. 各行の標準偏差を計算
apply(x, 1, sd)
```

```
##        a        b        c
## 1.290994 1.290994 1.290994
```

3.7.2 R のリスト形式 (List)

異なる型のデータをまとめる

- **リスト**: 数値、文字列、データフレームなど、異なる種類のオブジェクトを一つにまとめたもの。
- 分析関数の結果（例：k-means の結果）は、多くの場合リスト形式で返されます。

```
# 1. リストの作成
my_list <- list(numbers = 1:3,
                text = c("apple", "orange"),
                df = data.frame(a = 1:2, b = 3:4))
```

異なる型のデータをまとめる

```
# 2. 要素の取り出し方 (3 つの方法)
my_list$numbers # 名前で指定
```

```
## [1] 1 2 3
```

```
my_list[[2]]    # 番号で指定
```

```
## [1] "apple"  "orange"
```

```
my_list[["df"]] # 文字列で指定
```

```
##   a b
## 1 1 3
## 2 2 4
```

3.7.3 k-means 法 (k-Means Algorithm)

アルゴリズムの仕組み

1. グループ数 k を決める。
2. ランダムに k 個の中心点（セントロイド）を決める。
3. 各データを、最も近い中心点のグループに割り当てる。
4. グループごとに新しい中心点（平均）を計算する。
5. 中心点が動かなくなるまで 3 と 4 を繰り返す。

congress データの読み込み (1)

- 第 80 議会と第 112 議会の思想スコアを、行列形式にまとめます。

```
# 1. ローカルに保存したデータの読み込み（推奨）
congress <- read.csv("congress.csv")

#（参考）URL から直接読み込むことも可能
# congress <- read.csv("https://ayumu-tanaka.github.io/QSS/QSS_Data/congress.csv
```

議会データの準備 (2)

```
# 第 80 議会の 2 つの次元のスコアを結合 (cbind) して行列にする
dwnom80 <- cbind(congress$dwnom1[congress$congress == 80],
                 congress$dwnom2[congress$congress == 80])

# 第 112 議会も同様に作成
dwnom112 <- cbind(congress$dwnom1[congress$congress == 112],
                  congress$dwnom2[congress$congress == 112])
```

k-means の実行

- $k = 2$（2 つのグループ）として実行します。

```
# 1. k-means の実行
# centers = 2 (グループ数), nstart = 5 (ランダムな初期値を変えて 5 回試行)
k80.out <- kmeans(dwnom80, centers = 2, nstart = 5)
k112.out <- kmeans(dwnom112, centers = 2, nstart = 5)

# 2. 結果（中心点）の確認
k80.out$centers
```

```
##          [,1]       [,2]
## 1 -0.04843704  0.7827259
## 2  0.14681029 -0.3389293
```

分類結果と政党の比較（1）

- アルゴリズムが分けたグループが、実際の政党とどれくらい一致しているか確認します。

```
# 1. 第 80 議会 (1940 年代)
table(party = congress$party[congress$congress == 80],
      cluster = k80.out$cluster)
```

```
##              cluster
## party           1   2
##   Democrat    132  62
##   Other         0   2
##   Republican    3 247
```

分類結果と政党の比較（1）

```
# 2. 第 112 議会 (2010 年代)
table(party = congress$party[congress$congress == 112],
      cluster = k112.out$cluster)
```

```
##              cluster
## party           1   2
##   Democrat      0 200
##   Republican  242   1
```

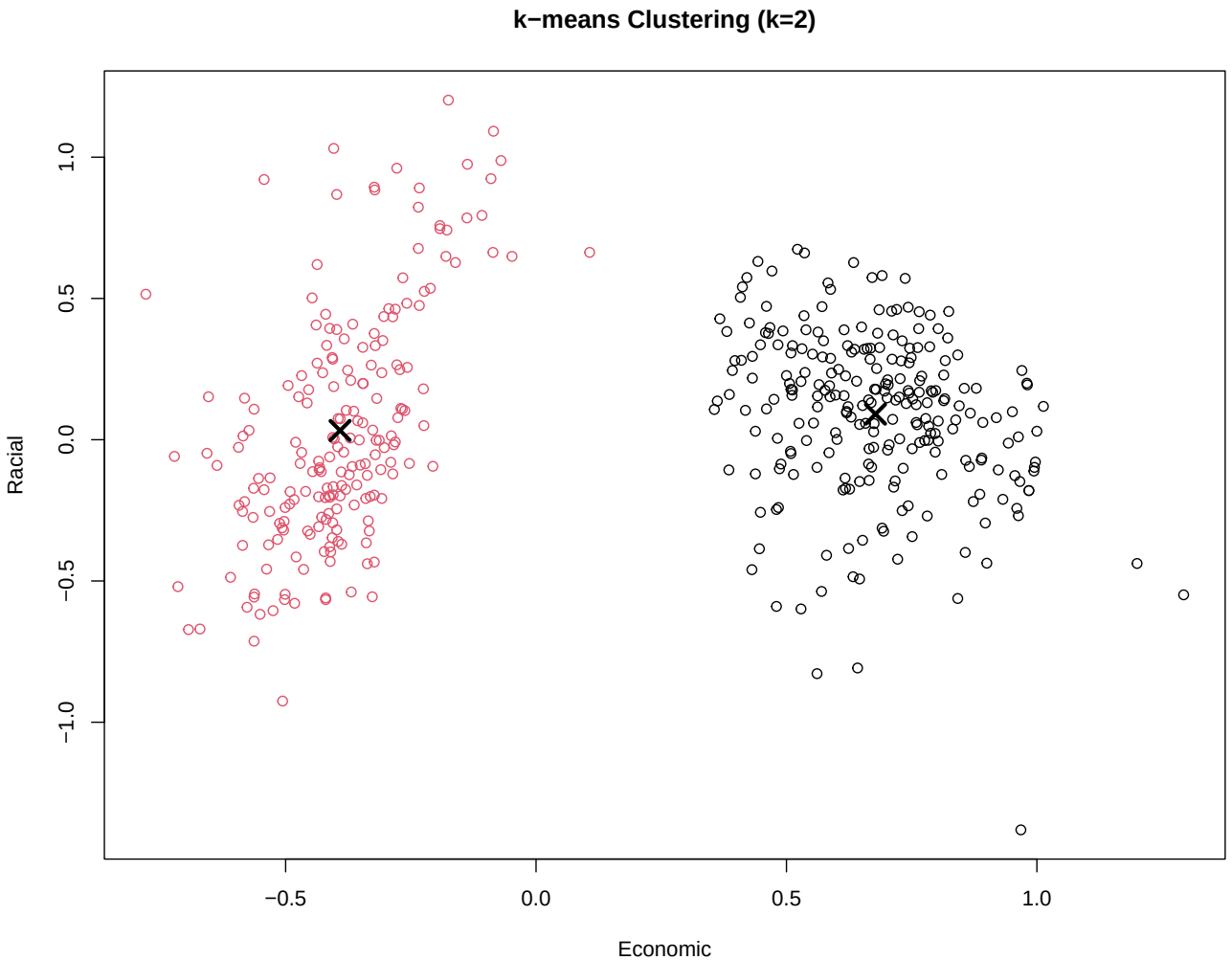
- **観察**: 第 112 議会では、クラスターと政党がほぼ完全に一致しており、政党間の思想的隔たり（分極化）が非常に明確であることがわかります。

クラスターの視覚化: コード

```
# クラスターごとに色を変えてプロット
# cluster 列 (1 or 2) をそのまま色として使用
plot(dwnom112, col = k112.out$cluster,
     xlab = "Economic", ylab = "Racial",
     main = "k-means Clustering (k=2)")

# 中心点 (centers) を × 印 (pch=4) で追加
points(k112.out$centers, pch = 4, cex = 2, lwd = 3)
```

クラスターの視覚化: 描画結果



3.7.4 まとめ

このセクションのまとめ

- **クラスター分析**: データの類似性に基づいて、自動的にグループを生成する。
- **k-means 法**: 中心点からの距離を最小化するようにグループを最適化する手法。
- **政治分析への応用**: 議員の思想スコアを分類することで、現代の米国議会が政党ごとにきれいに分極化していることを確認できた。
- **R の操作**:
  - `matrix()` と `apply()` による行列計算。
  - `list()` と `[[]]` による複雑なデータの管理。
  - `kmeans()` によるアルゴリズムの実行と、`table()` による結果の検証。
  - `points()` を使った中心点のプロット。