クレジット:
UTokyo Online Education 統計データ解析 Ⅱ 2018 小池祐太

東京大学 THE UNIVERSITY OF TOKYO

CC BY NC ND

統計データ解析 II (平成３０年度)

東京大学大学院数理科学研究科
統計データ解析教育研究グループ

村田　昇 (早稲田大学, 東京大学)
吉田朋広 (東京大学)
小池祐太 (東京大学)

第 9 章 クラスター分析

9.1. 目的

クラスター分析 *(cluster analysis)* とは, 主成分分析と並ぶ教師なし学習の代表的な手法の一つであり, 多数の個体に対するいくつかの共変量 (特徴量) の観測データが与えられたとき, それらの個体の間に隠れているクラスター構造 (グループ構造) を共変量の値に基づいて発見することを目的とする分析手法である. 同じクラスターに属する個体どうしは (なんらかの意味で) 近い性質をもち, 異なるクラスターに属する個体どうしは異なる性質をもつような少数のクラスターを見いだすことで, さらなるデータ解析やデータの可視化に役立てる目的で利用される.

クラスター分析には大きく分けて以下の 2 つのアプローチがある:

階層的方法: データ点およびクラスターの間に (共変量から定まる) 距離 (非類似度) を定義し, 近いものから順にクラスターを形成, もしくは近いものどうしがクラスター内に残るように分割しながら, グループ化していく方法

非階層的方法: クラスターの定め方の「良さ」を評価するための損失関数を定め (値が小さいほど良いとする), その損失関数を最小化するようにクラスターを形成して変数をグループ化していく方法

この資料では非階層的手法の代表的方法である k-**平均法** *(k-means clustering)* について説明する. なお, 階層的クラスタリングのうち凝縮的方法を実行するための関数 `hclust()` が R には用意されている. 使い方については `hclust()` のヘルプファイルを参照してほしい.

9.2. k-平均法

p 個の変数 $X_1, X_2, \ldots, X_p$ を n 個の個体について観測した観測データ $x_{i1}, x_{i2}, \ldots, x_{ip}$ $(i = 1, 2, \ldots, n)$ が与えられているとし, i 番目の個体に対する観測データに対応するベクトルを $\boldsymbol{x}_i = (x_{i1}, x_{i2}, \ldots, x_{ip})$ とする. クラスターの定め方は, 各個体番号 $i = 1, 2, \ldots, n$ に対してその個体が属するクラスター番号 $C(i)$ を定める対応 C として定式化できる. そのため, 非階層的クラスタリングは, このような対応 C の「良さ」を評価する損失関数を観測データ $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n$ を決めたのち, その損失関数を C に関して最小化するようにクラスターを定めることで実行できる.

k-平均法では, 最終的に得たいクラスターの個数 k をあらかじめ指定する. また, 2 つの個体 i, i' の「近さ」を共変量の観測データ間のユークリッド距離の二乗

$$\|\boldsymbol{x}_i - \boldsymbol{x}_{i'}\|^2 := \sum_{j=1}^{p} (x_{ij} - x_{i'j})^2$$

で評価する. そして, 同じクラスターに属する個体どうしが近いほど値が小さくなるように, C の損失関数 $W(C)$ を定める. 具体的には

$$W(C) := \sum_{l=1}^{k} \frac{1}{n_l} \sum_{i:C(i)=l} \sum_{i':C(i')=l} \|\boldsymbol{x}_i - \boldsymbol{x}_{i'}\|^2$$

と定義する. ただし, n_l は l 番目のクラスターに属する個体の総数を表す. いま, l 番目のクラスターに属する個体の特徴量の共変量の平均を

$$\bar{\boldsymbol{x}}_l := \frac{1}{n_l} \sum_{i:C(i)=l} \boldsymbol{x}_i$$

で定めると, 簡単な計算によって $W(C)$ は次のように書き直せる:

$$W(C) = 2 \sum_{l=1}^{k} \sum_{i:C(i)=l} \|\boldsymbol{x}_i - \bar{\boldsymbol{x}}_l\|^2.$$

従って, $W(C)$ を最小化するように C を定めることは, クラスター内変動の総和が最小になるようにクラスターを定めることと同等である.

さて, $W(C)$ が最小になるように C を定めるのが我々の目的である. C の取り方は k^n 通りで有限個のパターンしかないので, 原理的にはこの k^n 通り全てのパターンについて $W(C)$ の値を計算し比較することで, $W(C)$ を最小化する C が決定できる. しかし, サンプル数 n が十分小さくない限り, この方法は計算量の観点から現実には実行が不可能である. そのため, 現実的な計算量で実行可能であるような $W(C)$ の最小化のためのアルゴリズムがいくつか提案されているが, 代表的なものとして **Lloyd-Forgy のアルゴリズム**がある. Lloyd-Forgy のアルゴリズムでは, $\bar{\boldsymbol{x}}_l$ が

$$\sum_{i:C(i)=l} \|\boldsymbol{x}_i - \boldsymbol{\mu}_l\|^2$$

を最小化するような p 次元ベクトル $\boldsymbol{\mu}_l$ と一致することに着目して, C と $\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \ldots, \boldsymbol{\mu}_k$ の更新を以下の手順で繰り返すことで $W(C)$ を最小化する C を求める:

1. p 次元ベクトルの初期値 $\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \ldots, \boldsymbol{\mu}_k$ を与える
2. 各データ点 $i = 1, 2, \ldots, n$ について, $\|\boldsymbol{x}_i - \boldsymbol{\mu}_l\|$ を最小化するような l を i が所属するクラスター番号 $C(i)$ として定める
3. 各 $l = 1, 2, \ldots, k$ について, ベクトル $\boldsymbol{\mu}_l$ を

$$\boldsymbol{\mu}_l = \frac{1}{n_l} \sum_{i:C(i)=l} \boldsymbol{x}_i$$

   によって更新する
4. 平均ベクトルが更新前と更新後で変化しなかった場合計算を終了する. そうでなければステップ 2 に戻る

Lloyd-Forgy のアルゴリズムの成否は初期値のベクトル $\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \ldots, \boldsymbol{\mu}_k$ の選び方に依存するため, 応用上は複数の初期値の候補をランダムに試して, $W(C)$ の値を最も小さくする解を最終的な解として採用するということが行われる.

9.3. R での実行

R には k-平均法を実行するための関数 `kmeans()` が実装されている.

- クラスターの数 k はオプション `centers` で指定する.
- オプション `algorithm` では $W(C)$ を最適化するために利用するアルゴリズムが指定できる. デフォルトでは Lloyd-Forgy のアルゴリズムの改良版である Hartigan-Wong のアルゴリズムを利用する.
- オプション `nstart` では試す初期値の候補の数が指定できる.

なお, $W(C)$ の定義から明らかなように, 共変量のうちの 1 つを定数倍 (例えば測定値の単位を変更) すると, クラスタリングの結果が変わりうることに注意する必要がある. すべての共変量を同じスケールで評価したい場合は, 主成分分析の場合と同様に, 実行前にデータを標準化すればよい.

```
> ## 例 1: データセット iris に対する k-平均法
> ## クラスラベル Species を他の変数から正確に予測できるか検証
> x <- subset(iris, select = -Species) # 共変量の抽出
> # k-平均法の実行
> set.seed(123)
> out <- kmeans(x, centers = 3, nstart = 20) # k=3, 初期値は 20 通り試す
> # 結果の確認
> table(iris$Species[out$cluster == 1]) # setosa
    setosa versicolor  virginica
        50          0          0
> table(iris$Species[out$cluster == 2]) # virginica
    setosa versicolor  virginica
         0          2         36
> table(iris$Species[out$cluster == 3]) # versicolor
    setosa versicolor  virginica
         0         48         14
> # k-平均法の実行: データを標準化した場合
> set.seed(123)
> out <- kmeans(scale(x), centers = 3, nstart = 20)
> # 結果の確認
> table(iris$Species[out$cluster == 1]) # setosa
    setosa versicolor  virginica
        50          0          0
> table(iris$Species[out$cluster == 2]) # virginica
    setosa versicolor  virginica
         0         11         36
> table(iris$Species[out$cluster == 3]) # versicolor
    setosa versicolor  virginica
         0         39         14
> ### この場合, 標準化せずにスケールの情報をクラスタリングに
> ### 取り込んだ方が良好な結果が得られるようである
> # k-平均法の実行: k=2 としてみた場合
> set.seed(123)
> out <- kmeans(x, centers = 2, nstart = 20)
> # 結果の確認
> table(iris$Species[out$cluster == 1]) # setosa
    setosa versicolor  virginica
        50          3          0
> table(iris$Species[out$cluster == 2]) # virginica, versicolor
    setosa versicolor  virginica
         0         47         50
> ## 例 2: kendata.csv (6 章参照)
> ## 総務省統計局の統計データ
> ## http://www.stat.go.jp/data/shihyou/naiyou.htm
> ## 社会生活統計指標－都道府県の指標－ 2017 社会生活統計指標 2017 年 2 月 17 日公表
> ## http://www.e-stat.go.jp/SG1/estat/List.do?bid=000001083999&cycode=0
> ## 森林面積割合 Ratio of forest area (%) 2014
> ## 就業者 1 人当たり農業産出額 (販売農家)
> ## Gross agricultural product per agricultural worker (commercial farm households)
> ## (万円:10 thousand yen) 2014
> ## 全国総人口に占める人口割合
> ## Percentage distribution by prefecture (%) 2015
> ## 土地生産性（耕地面積 1 ヘクタール当たり）
> ## Land productivity (per hectare of cultivated land area)
> ## (万円:10 thousand yen) 2014
> ## 商業年間商品販売額［卸売業＋小売業］(事業所当たり)
> ## Annual sales of commercial goods        [wholesale and retail trade] (per establishment)
```

```
> ## (百万円:million yen) 2013
> kendata <- read.csv(file="kendata.csv",row.names=1,header = TRUE) # データの読み込み
> # k-平均法の実行:「八地方区分」を考慮して k=8 としてみる
> set.seed(123)
> out <- kmeans(scale(kendata), centers = 8, nstart = 20)
> # 結果の確認
> nam <- rownames(kendata) # 個体名
> nam[out$cluster == 1] # 北関東, 静岡, 香川, 北部九州の多く, 沖縄
[1] "Ibaraki"  "Tochigi"  "Gumma"    "Shizuoka" "Kagawa"   "Saga"     "Nagasaki"
[8] "Kumamoto" "Okinawa"
> nam[out$cluster == 2] # 東北・中部の一部, 近畿・中国地方の多く, 愛媛・大分
 [1] "Aomori"    "Iwate"     "Yamagata"  "Nagano"    "Gifu"      "Mie"
 [7] "Kyoto"     "Hyogo"     "Nara"      "Tottori"   "Shimane"   "Okayama"
[13] "Hiroshima" "Ehime"     "Oita"
> nam[out$cluster == 3] # 南九州
[1] "Miyazaki"  "Kagoshima"
> nam[out$cluster == 4] # 東北の一部, 北信越の多く, 滋賀・山口
[1] "Miyagi"    "Akita"     "Fukushima" "Niigata"   "Toyama"    "Ishikawa"
[7] "Fukui"     "Shiga"     "Yamaguchi"
> nam[out$cluster == 5] # 南四国, 山梨・和歌山
[1] "Yamanashi" "Wakayama"  "Tokushima" "Kochi"
> nam[out$cluster == 6] # 北海道
[1] "Hokkaido"
> nam[out$cluster == 7] # 首都圏 + 大都市
[1] "Saitama"  "Chiba"    "Kanagawa" "Aichi"    "Osaka"    "Fukuoka"
> nam[out$cluster == 8] # 東京
[1] "Tokyo"
```

(kmeans.r)

9.4. 参考文献

1. T. Hastie, R. Tibshirani, J. Friedman 著「The Elements of Statistical Learning」, Springer (2009 年).
2. G. James, D. Witten, T. Hastie, R. Tibshirani 著「An Introduction to Statistical Learning」, Springer (2013 年).
3. 金明哲著「R によるデータサイエンス (第 2 版)」, 森北出版 (2017 年).