x-means法改良の一提案

— k-means法の逐次繰り返しとクラスターの再併合 —

石岡 恒憲*

要 旨

k-means法の逐次繰り返しとBICによる分割停止基準を用いることで, クラスター数を自動的に決定するアルゴリズムx-means法を改良した. その手続きは, 分割順序に起因する好ましくないと考えられる分割クラスターを併合するものである. この併合操作により, さまざまな事例に対して, 適当と考えられるクラスター数を得ることのできる事例の数が大幅に増加することが確認された. この方法は, クラスター数未知のときに発見的な方法に拠らずに情報理論的に最適と考えられるクラスター数を求めることができる. その計算量(computational complexity)は標本サイズをN, クラスター数をkとしたとき, $\mathcal{O}(N \log k)$ となる.

1. はじめに

クラスタリングは, データマイニング分野における重要なツールの1つである(Fayyad *et al.*, 1996). データマイニングを大量のデータからの知識発見と捉えるならば, データ処理の可用性, また計算量の問題の他に, データ全体が果たしていくつのクラスターに分割されるかというクラスターの数それ自体が有用な情報となる.

クラスター数未知における, 最適クラスターを求める方法には, 大別して以下の4つがある.

1. クラスター数の異なる幾通りかのクラスタリングに対して, 情報量規準などの適当な規準を用いて最適なクラスターを求める方法.
2. 最小体積楕円体(minimum volume ellipsoid; MVE)推定量を用いる方法(Jolion *et al.*, 1991).
3. 最適解より多めのクラスター分割から始めて, 近いクラスター同士を併合したり, 疑似(spurious)のクラスターを抹消することにより, 適当な数のクラスターを決定する方法(Krishnapuram & Freg, 1992).
4. 最初にある十分小さい数のクラスターにk-means法で分類した後, 各クラスターに対して, 同様にk-means法による2分割を, その分割が適当でないと判断されるまで繰り返す方法(Pelleg & Moore, 2000).

1.は最もシンプルな方法であるが, 一般に情報量規準などの評価関数がクラスター数に対して単峰とならない. また大量データの場合は, 理論的にはクラスター数の異なる大量のクラスタリングが可能であるために, これらを総当たりで評価する必要があり, 計算量の問題から現実的でない. なお, クラスター数の決定基準については, Hardy(1996)が最適なクラスターを選択するために提案された代表的な7方法（うち2つは階層的なクラスタリングについてのみ適用可能）に対して

* 独立行政法人 大学入試センター 研究開発部 試験作成支援研究部門, 〒153-8501 東京都目黒区駒場2-19-23, E-mail: tunenori@rd.dnc.ac.jp

さまざまな事例でもってその比較・評価を与えている.

2.は各クラスターは楕円体で分割されるという仮定のもとで, 全体から始めKolmogrov-Smirnov検定を用いて, 単一の楕円体とみなすことのできるクラスターを順次, 決定し, 取り除いてゆくものである. しかしながら, この方法は, 検定を行う際の有意水準の決定が難しいこと, またデータの外乱に極めて弱いことが知られている(Nasraoui *et al.*, 2005).

3.では全てのデータを排他的に分類するのではなく, 外れ値と見なされるデータはクラスターから除外されるものである.

4.は最終的に出来上がるクラスターの数が不定であることから, x-means法と呼ばれている. その基礎となるk-means法については, その計算量の少なさ故に, 大量データへの適用や高速化の研究が精力的に進んでいる. 例えばPelleg & Moore(1999)はkdツリー(Bentley, 1980)のノードに必要な情報を格納することで, クラスター重心を更新する計算量を大幅に減じている. Huang(1998)は大規模なカテゴリカル・データに対する高速化手法を提示している. Zhang *et al.*(1996)の提案するBIRCHは, 最初にデータ全体を走査してClustering Feature Tree(CF-tree)というデータの要約情報を生成し, 以降の操作をこのCF-treeだけに限定するものである. このCF-tree はNに比べて十分小さいために, 最初と最後にデータを走査するための計算量$\mathcal{O}(N)$が全体の計算量となり, 主記憶上に保持できるという点で注目すべき研究である.

このようにk-means法の研究が進んでいることから, 著者はさきに4.の方法を支持し, さらに以下の点について改良を加えた(石岡, 2000) .

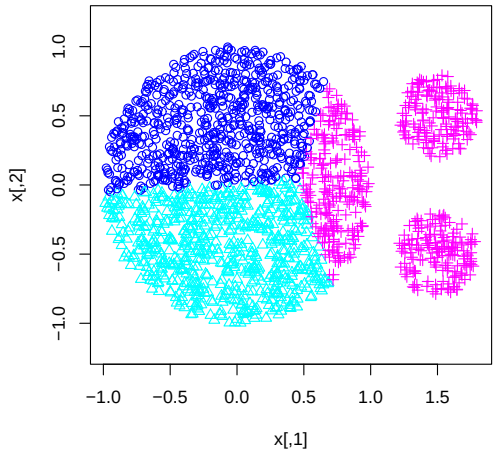
1. 逐次分割されるサブクラスターごとに重心廻りの分散の違いを考慮した. これは本質的な改良である.
2. 対数尤度の計算の一部に近似計算を用い, また2分割手続きにおいても逐次分割に関数の再帰呼出しを用いるのではなく, 2分割のうちの一方に対してのみ分割を継続し, 他方は一旦スタックに積んでおき, 後で処理することで, 逐次分割の階層が深くなったときの関数呼び出しのオーバヘッドを大幅に低減した.

このアルゴリズムをフリーソフトであるRで実装し, それが様々なデータ事例でもって有効に機能することを示した(石岡, 2000; Ishioka, 2000). しかしながら, k-means法は各クラスターの形状が超球状でクラスター内のデータ数がどれもほぼ等しいという仮定を暗黙のうちに仮定しているので, この仮定に反する構造の抽出は困難である. Guha *et al.*(1998)では, 図1(a)のような3つの密な領域が存在するデータに対してk-means法を適用した結果を例示している. 直観に反して, データ数の最も大きな領域は3つに分割されている. またデータ数の比較的小さい2つの領域は, 2つに分割されていない.

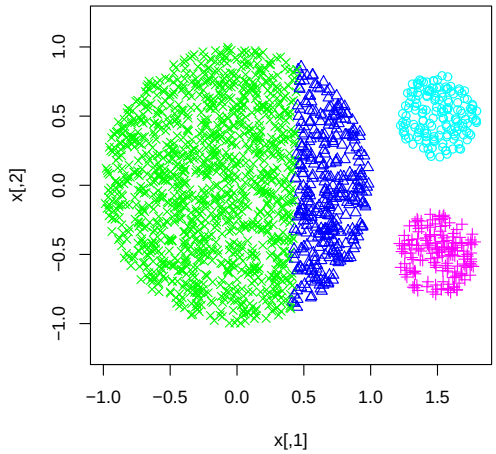
この同じデータに対して初期分割$k_0 = 2$としてx-meansを適用した結果が図1(b)である. 全体は4つのクラスターに分類される. データ数の最も大きな領域はk-means法のもつ欠点をひきずって2つに分割されるが, データ数の小さな2つの領域は直観通りに2つに分割されることがわかる.

このようにx-means法はk-means法のもつ欠点をある程度緩和するものの, その欠点は完全には解消されない. しかしながら, データ数の最も大きい領域に対して, 2つに分割されるよりも1つのクラスターと見なした方が情報理論におけるエントロピーが増大するという意味で好ましいと判断し, 1つに併合するする手続きが加えられるなら, より適当なクラスタリングが可能となるであろう.

そこで本稿では, x-meansにおいてクラスター分割をした後に併合操作を付加する手続きを提案する. またこの操作が, さまざまなシミュレーショ



(a) k-means ($k = 3$)



(b) x-means ($x = 4$)

図 1: 分割アルゴリズムによるクラスタリング

ン実験で有効に機能することを示す.

2節では, x-means法のアルゴリズムについて概説し, 今回, 加える操作について説明する. 3節では, その結果, 生成されるクラスター数について評価し, 計算量について考察する. 4節はまとめである.

2. x-means法

2.1 x-means法の概説

石岡(2000)の提案するアルゴリズムは, 以下のように要約される.

0. 解析すべきデータとしてn個のp次元データを用意する.

1. 十分に小さなクラスター数の初期値k_0 （特に指定しなければ2）を定める.

2. $k = k_0$としてk-meansを適用する. 分割後のクラスターを

$$C_1, C_2, \ldots, C_{k_0}$$

とする.

3. $i = 1, 2, \ldots, k_0$とし, 手順4〜9を繰り返す.

4. クラスターC_iに対して$k = 2$としてk-meansを適用する. 分割後のクラスターを

$$C_i^1, C_i^2$$

とする.

5. C_iに含まれるデータ$\boldsymbol{x}_i$にp変量正規分布

$$f(\boldsymbol{\theta}_i; \boldsymbol{x}) = (2\pi)^{-p/2} |\boldsymbol{V}_i|^{-1/2} \exp\left[-\frac{1}{2}(\boldsymbol{x} - \boldsymbol{\mu}_i)^t \boldsymbol{V}_i^{-1} (\boldsymbol{x} - \boldsymbol{\mu}_i)\right]$$

を仮定し, そのときのBIC(Baysian Information Criterion; Schwarz, 1978)を以下により計算する:

$$\mathrm{BIC} = -2 \log L(\widehat{\boldsymbol{\theta}_i}; \boldsymbol{x}_i \in C_i) + q \log n_i$$

ここに$\widehat{\boldsymbol{\theta}_i} = [\widehat{\boldsymbol{\mu}_i}, \widehat{\boldsymbol{V}_i}]$ は, p変量正規分布の最尤推定値とする; $\boldsymbol{\mu}_i$はp次の平均値ベクトル, $\boldsymbol{V}_i$は$p \times p$の分散・共分散行列である; qはパラメータ空間の次元数で, $\boldsymbol{V}_i$の共分散を無視すれば（0と置けば）, $q = 2p$である. 共分散を無視しなければ, $q = p(p+3)/2$である. $\boldsymbol{x}_i$はクラスターC_iに含まれるp次元データとし, n_iはC_iに含まれるデータ数とする. Lは尤度関数で$L(\cdot) = \prod f(\cdot)$である.

共分散を無視する, すなわち共分散行列を対角行列とみなし, 対角成分の数をパラメータ数とすることは, 計算の簡略化のための簡便

法である．しかし, 共分散を無視する場合は, その仮定が実際のデータに適合しているものか考慮する必要がある．例えば多次元尺度法や主成分分析を用いて, もとのデータベクトル空間とは異なった, 共変量の影響の少ないベクトル空間に対して本アルゴリズムを適用するのであれば, 共分散を無視してもその影響は少ないものと考えられる．

6. C_i^1, C_i^2 のそれぞれに対して，パラメータ $\boldsymbol{\theta}_i^1, \boldsymbol{\theta}_i^2$ をもつp変量正規分布を仮定し，2分割モデルにおいてデータの従う確率密度を

$$\boldsymbol{x}_i \sim \alpha_i [f(\boldsymbol{\theta}_i^1; \boldsymbol{x})]^{\delta_i} [f(\boldsymbol{\theta}_i^2; \boldsymbol{x})]^{1-\delta_i} \quad (2.1)$$

とおく．ここで

$$\delta_i = \begin{cases} 1, \ \boldsymbol{x}_i \text{が} C_i^1 \text{に含まれるとき} \\ 0, \ \boldsymbol{x}_i \text{が} C_i^2 \text{に含まれるとき} \end{cases}$$

とする．（$\boldsymbol{x}_i$はC_i^1かC_i^2のいずれか一方に必ず含まれる．）またα_iは, (2.1)式を確率密度とするための基準化定数であるが, その近似として,

$$\alpha_i = 0.5/K(\beta_i)$$

により計算する．　ここで，$K(\cdot)$は標準正規分布の下側確率とする．β_iは$f(\boldsymbol{\theta}_i^1; \boldsymbol{x}_i)$と$f(\boldsymbol{\theta}_i^2; \boldsymbol{x}_i)$の分離の程度を示す指標で

$$\beta_i = \sqrt{\frac{\|\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2\|^2}{|\boldsymbol{V}_1| + |\boldsymbol{V}_2|}}$$

で示すものとする．

(2.1)式は信頼性工学や医学統計の分野では競合正規分布モデルと呼ばれているものであるが, このモデルにおいて β_iはクラスター決定(どちらの正規分布に属するか)の指標を与えるものではない．

この2分割モデルにおけるBICを以下により計算する:

$$\mathrm{BIC}' = -2 \log L(\widehat{\boldsymbol{\theta}_i'}; \boldsymbol{x}_i \in C_i) + q' \log n_i$$

ここに$\widehat{\boldsymbol{\theta}_i'} = [\widehat{\boldsymbol{\theta}_i^1}, \widehat{\boldsymbol{\theta}_i^2}]$ は, 2つのp変量正規分布の最尤推定値である．　共分散を無視すれば, 各pに対し平均と分散の2つのパラメータが存在するので, パラメータ空間の次元は$q' = 2 \times 2p = 4p$となる．共分散を無視しなければ, $q' = 2q = p(p+3)$である．

7. BIC $>$ BIC$'$ならば, 2分割モデルをより好ましいと判断し, 2分割を継続すべく

$$C_i \leftarrow C_i^1$$

とする．C_i^2については, p次元データ, クラスターの重心, 対数尤度とBICを保持し，これらをスタックに積む．手順4へ．

8. BIC $\leq$ BIC$'$ならば, 2分割しないモデルをより好ましいと判断し, C_i^1についての2分割を停止する．（手順7で作成された）スタックからデータを取り出し,

$$C_i \leftarrow C_i^2$$

とし, 手順4へ．スタックが空なら次の手順へ．

9. C_iにおける2分割が全て終了．手順4〜8で作成された2分割のクラスターがC_i内で一意になるようにデータの属するクラスター番号を振りなおす．

10. はじめにk_0分割したクラスター全てについて2分割が終了．全データに対してそれらの属するクラスター番号が一意になるように番号を振りなおす．

11. 全データの属するクラスター番号, および各クラスターの重心, 各クラスターに含まれるデータ数を出力する[終了]

モデル選択規準として提案されている多くの情報量規準の中からBICを用いるのは, 以下の理由による．

- BICがその導出過程で, 指数型分布族の選択を考えていること（正規分布は指数型分布族に含まれる）
- 分布間の距離に基づくのではなく, モデルの事後確率を比較していること

2.2 x-means法の改良

図1(b) の例において, 最も大きな領域が2つに分割されてしまうのは, 最初の分割でデータ数がほぼ同じになるように全体を分割してしまうことによる. もし, この最も大きい領域が2つに分割されるのと1つにまとめるのとどちらが好ましいかと問うならば, 直観的に後者, すなわち1つにまとめた方を選ぶであろう. それ故, 手順10.と手順11.の間に以下の手順を追加することが提案される.

10-2. C_iに含まれるデータ数をn_iとし, n_iを小さい順に並べる. このとき添字iの並びを$\boldsymbol{I} = (I_1, I_2, \ldots, I_k)$と置く. Iの添字を改めて$i, j(= 1, \ldots, k)$とすれば, $i < j$のとき$n_i < n_j$である.

$i < j$ なる全てのi, jの組合せに対して, C_iに対するBICと, C_iとC_jに対するBIC′を比較する. もしBIC $>$ BIC′ならC_iをC_jに併合する. ただし併合操作は任意のi, jに対し, 1回限りとする.

例として$n_1 = 15,\ n_2 = 50,\ n_3 = 10$としたときの手順は以下のようになる. n_iを小さい順に並べると, $n_3 < n_1 < n_2$であるから, $\boldsymbol{I} = (3, 1, 2)$である. 従って

$$(i, j) = (3, 1),\ (3, 2),\ (1, 2)$$

の順に評価する.

始めに$(i, j) = (3, 1)$について評価する. ここで

$$(C_3\text{に対するBIC}) > (C_3\text{と}C_1\text{に対するBIC}) \tag{2.2}$$

の関係が成り立てば, C_3をC_1に併合する. 次に$(i, j) = (3, 2)$について評価するが, もし, 既に(2.2)式の関係がなりたっていれば, C_3は既にC_1に併合されているので, この操作をスキップする(評価しない). そうでなければ, 次に

$$(C_3\text{に対するBIC}) > (C_3\text{と}C_2\text{に対するBIC}) \tag{2.3}$$

の関係について評価し, 成立すればC_3をC_2に併合する. 最後に$(i, j) = (1, 2)$について評価する. もし, 既に(2.2)式あるいは(2.3)式の関係がなりたっていれば, C_1あるいはC_2は既に併合操作を実行しているので, この操作をスキップする. 要するに併合されたクラスターに更なる併合を行わない.

3. 性能評価

3.1 生成されるクラスター数に関する調査

(1) 図1のデータを以下の手順で作成し, シミュレーションにより評価する.

$u_1 \sim \mathrm{Unif}(-1, 1), u_2 \sim \mathrm{Unif}(-1, 1)$とし, $u_1^2 + u_2^2 \leq 1$を満たす(u_1, u_2)の組を$\boldsymbol{u}_j (j = 1, \ldots 2000)$とする. この$\boldsymbol{u}_j$を以下の通り変換し, $\boldsymbol{x}_j$を得る.

$$\begin{aligned}
&\boldsymbol{x}_j = \boldsymbol{u}_j,\ (j = 1, \ldots, 1600)\\
&\boldsymbol{x}_j = 0.3\boldsymbol{u}_j + [1.5, 0.5]',\ (j = 1601, \ldots, 1800)\\
&\boldsymbol{x}_j = 0.3\boldsymbol{u}_j = [1.5, -0.5]',\ (j = 1801, \ldots, 2000)
\end{aligned}$$

ここでUnifは一様乱数を, ′は転置を示す. この$\boldsymbol{x}_j$に対し, 初期分割$k_0 = 2$からはじめて, 逐次分割を繰り返すx-meansの操作を1,000回実施した. 上記の乱数は, その都度, すなわち1,000組を発生させた.

表 1 はx-meansによって得られたクラスターの数を要約したものである. 上段は併合操作のない従来のx-meansの結果である. 1,000回のシミュレーションの結果, 最頻値は4個のクラスターが得られた場合であり, その回数は 314回である. 2番目に多いのは5個のクラスターが得られた場合で

あり, 255回となる. 適当と考えられる3つに分割されることはただの1回としてなかった. これはx-means法が, 初期分割$k_0 = 2$において大きな領域を分割してしまうために, 結果としての3分割を得ることが困難であることによると考えられる.

下段は本稿で提案する併合操作を含むx-meansの結果である. 最頻値は適当と考えられる 3個のクラスターが得られた場合であるが, その回数は過半数の564回に及ぶ. 1,000回中564回という成績では, 十分に提案法の有効性を示していることにはならない, という謗りは甘受せざるを得ないが, 困難な条件, すなわち最も大きな領域が全体の8割を占めるような, また2分割の逐次分割では得ることが難しい全体が3つに分割された領域に対しても, 過半数が適切な解を与えたことはそれなりに評価してよいと考えられる. また提案法は従来法の結果を飛躍的に改善することがみてとれる. もっとも, 併合操作を1回に限らなければ, 提案法の結果はより改善されることが期待できる. しかしながらその場合は, 併合の順序をどのようにするか, は極めて本質的で難しい問題となる.

表 1: Guha *et al.*(1998)**の例にある3つに分割されるべきデータに対するクラスターの数**

x-means	2	3	4	5	6	7	8	9	10-13	計
従来法	14	0	314	255	250	106	37	17	7	1,000
提案法	13	564	303	90	23	6	1	0	0	1,000

なおk-meansのアルゴリズムは, Rに実装されているHartigan & Wong(1979)を用いた.

以下(2),(3),(4)では5群に分類されることが適当であると考えられる様々な事例に対して, 得られたクラスターの個数について評価する. x-means法では初期分割$k_0 = 2$からはじめて2分割を繰り返すわけだが, 素数である5群を選択できるかは興味深い実験であろう.

(2) クラスター中心が直線に並ぶように以下の2変量正規乱数を各50個, 計250個作成し, シミュレーションにより評価する.

$$\boldsymbol{x}_j \sim N(\boldsymbol{\mu} = [0,0]', \boldsymbol{\sigma} = [0.2, 0.2]'), \ (j = 1, \ldots, 50)$$
$$\boldsymbol{x}_j \sim N(\boldsymbol{\mu} = [-1,-1]', \boldsymbol{\sigma} = [0.2, 0.2]'), \ (j = 51, \ldots, 100)$$
$$\boldsymbol{x}_j \sim N(\boldsymbol{\mu} = [1,1]', \boldsymbol{\sigma} = [0.2, 0.2]'), \ (j = 101, \ldots, 150)$$
$$\boldsymbol{x}_j \sim N(\boldsymbol{\mu} = [2,2]', \boldsymbol{\sigma} = [0.2, 0.2]'), \ (j = 151, \ldots, 200)$$
$$\boldsymbol{x}_j \sim N(\boldsymbol{\mu} = [3,3]', \boldsymbol{\sigma} = [0.2, 0.2]'), \ (j = 201, \ldots, 250)$$

ここで$\boldsymbol{\mu}$は平均, $\boldsymbol{\sigma}^2$は分散を示す. 図 2はデータの一例である.

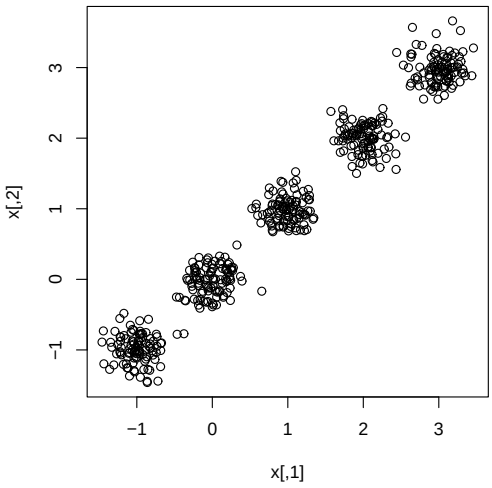


図 2: **クラスター中心が直線に並ぶ2変量正規乱数の例**

表 2 はx-meansによって得られたクラスターの数を要約したものである. 上段は併合操作のない従来のx-meansの結果である. 1,000回のシミュレーションの結果, 最頻値は5個のクラスターが得られた場合であり, その回数は 533回である. 2番目に多いのは6個のクラスターが得られた場合であり, 317回となる. 下段は本稿で提案する併合操作を含むx-meansの結果である. 最頻値は5個のクラスターが得られた場合であるが, その回数は実に 909回に及ぶ. 従来法は, 適当と考えられるクラスター数よりも多めに分割する傾向があったのだが, 提案法はその欠点を克服することがわかる.

k-meansでは, 求めたクラスターの各重心は, 必ずしも要素の凝集している場所に収束するとは限らない. このため従来法のx-meansでは, 新た

表 2: クラスター中心が直線に並ぶ250個の2変量正規乱数によって得られるクラスターの数

x-means	4	5	6	7	8	9	計
従来法	0	533	317	108	34	8	1,000
提案法	0	909	86	5	0	0	1,000

なクラスターの重心が要素の凝集する場所に収束するまで2分割を繰り返すことがしばしば起き, このために6つ以上のクラスターが得られる. 本実験でも, $50 \times 5 = 250$個のデータを最初の分割で約半分づつ（約125個づつ）に2分割したならば, それぞれのサブクラスターでこれを$50 + 50 + 25$の3つのクラスターに分けることが多く, 従って全体で6つのクラスターになる例が散見された. しかるに提案法では, この2つに分割された25+25個のクラスターを1つに併合することができる.

(3) クラスター中心が十字に配置されるように, 以下の300個の2変量正規乱数を生成する.

$$\boldsymbol{x}_j \sim N(\boldsymbol{\mu} = [0,0]', \boldsymbol{\sigma} = [0.2, 0.2]'), \ (j = 1, \ldots, 100)$$
$$\boldsymbol{x}_j \sim N(\boldsymbol{\mu} = [-2,0]', \boldsymbol{\sigma} = [0.3, 0.3]'), \ (j = 101, \ldots, 150)$$
$$\boldsymbol{x}_j \sim N(\boldsymbol{\mu} = [2,0]', \boldsymbol{\sigma} = [0.3, 0.3]'), \ (j = 151, \ldots, 200)$$
$$\boldsymbol{x}_j \sim N(\boldsymbol{\mu} = [0,2]', \boldsymbol{\sigma} = [0.4, 0.4]'), \ (j = 201, \ldots, 250)$$
$$\boldsymbol{x}_j \sim N(\boldsymbol{\mu} = [0,-2]', \boldsymbol{\sigma} = [0.4, 0.4]'), \ (j = 251, \ldots, 300)$$

5群のうち1つは, 100個の要素$(j = 1, \ldots, 100)$により構成され, 他の4つは各50個の要素により構成される. (2)で発生させる2変量正規乱数のクラスター中心が一直線上に並ぶのに対し, (3)ではクラスター中心は十字の形に配置され, しかも各クラスターにおける2変量正規乱数の分散は同一ではない. この場合の正規乱数の例を図 3に示す. 結果を表3に示す.

表 3: クラスター中心が十字に配置される2変量正規乱数300個によるクラスターの数

x-means	2	3	4	5	6	7	8	9	計
従来法	2	6	9	469	383	99	27	5	1,000
提案法	22	5	0	890	74	9	0	0	1,000

提案法では, 最も多い事例は5つのクラスターに

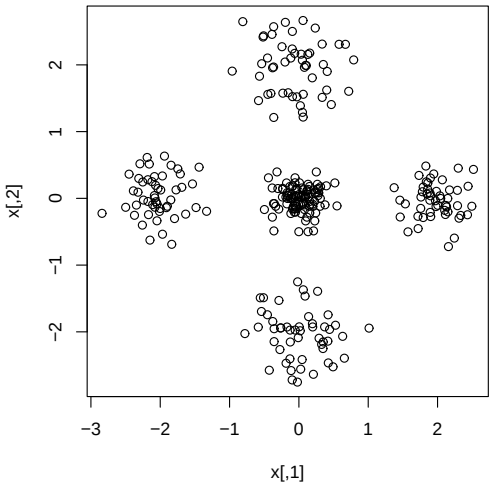


図 3: クラスター中心が十字に配置される2変量正規乱数300個の分類

分類される場合であり, その回数は890回である. 従来法では6つあるいはそれ以上のクラスターに分類される多くの事例が, 5つのクラスター分類の範疇に変換されることがわかる.

(4) クラスター中心が十字に配置され, 各群における2変量の共分散が0.5となるように300個の2変量正規乱数を生成する. 各群における平均と分散は(3)の例と同じとする. 図 4 には, このシミュレーションで用いた正規乱数の例を示す. 表 4には結果を示す.

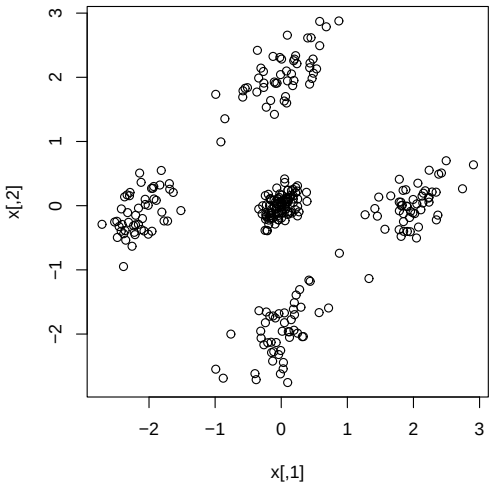


図 4: 相関係数0.5の2変量正規乱数の例

表 4: クラスター中心が十字に配置される相関係数0.5の2変量正規乱数300個によるクラスターの数

x-means	2	3	4	5	6	7	8	9	11-12	計
従来法	13	9	1	345	388	166	61	12	5	1,000
提案法	17	4	0	638	273	59	8	1	0	1,000

提案法では, 最も多い事例は5つのクラスターに分類される場合であり, その回数は638回である. 2番目に多い事例は, 6つのクラスターに分類される場合であり273回である. 適当と考えられるクラスタリングの行われる事例の数は(3)に比べ減少したが, (3)と同様の傾向, すなわち従来法では6つあるいはそれ以上のクラスターに分類される多くの事例が, 5つのクラスター分類の範疇に変換されることがみてとれる.

3.2 計算量に関する考察

まずk-means法の計算量が, クラスター数をk, 標本サイズをNとしたとき$\mathcal{O}(kN)$であることに注意する. x-means法はk-means法の逐次繰り返しであり, 最終的にk個のクラスターが発見されるので, 2分割木がバランスしていると仮定すれば, 標本サイズNに対して$k=2$のk-meansを1回, $N/2$に対して($k=2$のk-meansを)2回, $N/4$に対して4回実施することになり, 結局, 各階層におけるk-means計算量の総計はkに依らず$\mathcal{O}(N)$である. 一方, 2分割木における階層の深さは, 最下層のノード(クラスター)の個数が2^kであることから, ルートの階層深さを0とすれば, 平均的に$(\log k)/(\log 2)$である. これよりx-means法の計算量は, $\mathcal{O}(N\log k)$であることがわかる.

しかしながらkの大きさは, 通常の場合Nに比べ圧倒的に小さいので, $\log k$の大きさはさほど有利には働かない. むしろx-means法の計算量は, k-means法に比べ大きくなる. その理由としては以下が挙げられる.

1. 最適なクラスター分割が行われた後にも, 更に各クラスターに対して分割を行い, その分割が適当でないことを評価する必要がある. すなわち, 最適なクラスター数がkであるときに, $(2k)$個のクラスターに分割する作業を必然的に伴うこと. (クラスタリングの回数は$(2k-1)$である.)
2. 分割が妥当であるかを判断するための統計量規準としてBIC を計算する必要があること. このBICの計算量も, 共分散を無視した場合, 多変量正規分布の仮定ではデータの1次モーメントと2次モーメントから計算できるので, 計算量は$\mathcal{O}(N\log k)$であることがわかる.

上記で述べたことは, 実際に$k=5,10$とし, $N=10^3,10^4,10^5,10^6$ に変えたときのx-meansの計算時間(CPU時間)より今回あらたにわかったことである. なお追加した手順10-2の計算量については, 最大で$k(k-1)/2$回のBICの比較が必要となる. しかしながら, BICの値は逐次分割の際に既に計算されており, またkは標本サイズNに比べ十分小さいので, この計算量は全体に比べて無視できる.

4. おわりに

x-meansの2分割手続きによって生成されたクラスターを再評価し, 必要ならばそれらのクラスターを併合する手続きを加えることにより, x-meansの性能が格段に向上することが示された.

x-meansは基本的にはk-meansを逐次繰り返して使うものであるから, k-meansのより効果的な, あるいは実行速度の優れた方法が開発され, そのソースコードがCやFortranで入手できれば, k-meansの結果を容易に利用することができる.

我々のプログラムはRおよびFortran(g77)で実装されており, そのソースコードはhttp://www.rd.dnc.ac.jp/~tunenori/xmeans.html(日本語), またはhttp://www.rd.dnc.ac.jp/~tunenori/xmeans_e.html(英語) より入手することができる.

Fortranプログラムは tar+gzipで圧縮されており, これをダウンロードした後は, GNUの標準的な手順, すなわち `./configure; make; make install` でインストールすることができる. 本プログラムは, 標準的なlinuxで動作するg77コンパイラにおいて, $k = 10, N = 10^6$のデータに対して動作する. Windows上で動作する表計算ソフトExcelでは, 許容できるワークシートのサイズの上限は$N = 65,536$であるから, これに比較すれば, はるかに大量の(実に15倍の) データを取り扱うことのできることがわかる. もちろん, k-meansのアルゴリズムにBIRCHを用いれば, Nの上限は実質上, 無制限となる.

謝 辞

本稿の不備をご指摘いただいた2名の匿名の査読者, ならびに編集の労をとっていただいた栗原考次先生に厚くお礼申し上げます. なお, この研究の一部は文部科学省科学研究補助金 基盤研究(C) (課題番号16500628, 代表 石岡恒憲)の援助を受けた.

参考文献

Bentley, J. L. (1980). Multidimensional Divide and Conquer. *Communications of the ACM*, **23**(4), 214–229.

Fayyad, U. M., Piatetsky-Shapiro, G., & Smyth, P. (1996). From Data Mining to Knowledge Discovery: An Overview, in Fayyad, U. M., Piatetsky-Shapiro, G., Smyth, P. & Uthurusamy R. eds. *Advances in Knowledge Discovery and Data Mining,* chapter 1, 1–34, AAAI Press/The MIT Press.

Guha, S., Rastogi, R. & Shim, K. (1998). CURE: An Efficient Clustering Algorithm for Large Databases. in Proc. of *the ACM SIGMOD International Conference on Management of Data*, 73–84.

Hardy, A. (1996). On the Number of Clusters, *Computational Statistics & Data Analysis*, **23**, 83–96.

Hartigan, J.A. & Wong, M.A. (1979). A K-means Clustering Algorithm. *Applied Statistics*, **28**, 100–108.

Huang, Z. (1998). Extension to the k-Means Algorithm for Clustering Large Data Sets with Categorical Values. *Data Mining and Knowledge Discovery*, **2**(3), 283–304.

Ishioka, T. (2000). Extended K-means with an Efficient Estimation of the Number of Clusters. Intelligent Data Engineering and Automated Learning — IDEAL 2000, Second International Conference, Shatin, N.T., Hong Kong, China, December 2000, proceedings 17–22. (Lecture Notes in Computer Science 1983, Kwong Sak Leung, Lai-Wan Chan, Helen Meng (Eds.), Springer, 17–22, 2000) Available online: `http://www.rd.dnc.ac.jp/~tunenori/doc/`

Jolion, J. M., Meer, P. & Bataouche, S. (1991). Robust clustering with applications in computer vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **13**(8), 791–802.

Krishnapuram, R. & Freg, C. P. (1992). Fitting an unknown number of lines and planes to image data through compatible cluster merging. *Pattern recognition*, **25**, 385–400.

Nasraoui O., Leon E., & Krishnapuram R. (2005). Unsupervised Niche Clustering: Discovering an Unknown Number of Clusters in Noisy Data Sets. Chapter 8, 157–188, in *Evolutionary Computing in*

Data Mining, A. Ghosh & L. C. Jain, Eds, Springer Verlag.

Pelleg, D. & Moore, A. (1999). Accelerating Exact k-means Algorithms with Geometric Reasoning. *KDD-99*, 277-281.

Pelleg, D. & Moore, A. (2000). X-means: Extending K-means with Efficient Estimation of the Number of Clusters. *ICML-2000*, 727-734.

Schwarz, G. (1978). Estimating the Dimension of a Model. *Ann. Statist.*, **6**(2), 461-464.

Zhang, T., Ramakrishnan, R. & Livny, M. (1996). BIRCH: An Efficient Data Clustering Method for Very Large Databases. *SIGMOD Conf.*, 103–114.

石岡 恒憲 (2000). クラスター数を自動決定するk-meansアルゴリズムの拡張について, 応用統計学, **29**(3), 141–149. Available online: `http://www.rd.dnc.ac.jp/~tunenori/doc/`

AN EXPANSION OF X-MEANS

— PROGRESSIVE ITERATION OF K-MEANS AND MERGING OF THE CLUSTERS —

Tsunenori ISHIOKA*

* Dept. of Applied Statistics and Measurement, Research Division, The National Center for University Entrance Examinations, Komaba 2-19-23, Meguro-ku, Tokyo 153-8501, Japan

We expand a non-hierarchical clustering algorithm that can determine the optimal number of clusters by using iterations of k-means and a stopping rule based on BIC. The procedure requires merging the clusters that a k-means iteration has made to avoid unsuitable division caused by the division order. By using this additional merging operation, the case of adequate clustering was increased for various types of simulation runs. With no prior information about the number of clusters, our method can get the optimal clustering based on information theory instead of on a heuristic method. The computational complexity of our method is $\mathcal{O}(N \log k)$ for the sample size N and the number of final clusters, k.

Key words: clustering, data mining, unknown number of clusters