統計計算

# 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

大学入試センター　研究開発部　情報処理部門　石岡　恒憲

# Extended K-means with an Efficient Estimation of the Number of Clusters

Tsunenori ISHIOKA

## 1　はじめに

非階層的なクラスタリング方法の 1 つである k-means 法は，クラスター数を $k$，標本の大きさを $N$ としたとき $\mathcal{O}(kN)$ の計算量で済むために，自己組織化マップ (self-organizing map) とともに，特に大規模データのクラスタリングにしばしば用いられている（自己組織化マップを用いたクラスタリングについては Vesanto et al. (1999), Vesanto and Alhoniemi (2000), Yang and Ahuja (1999) など）．ことに近年のデータマイニング研究の隆盛にともない，k-means の高速化の手法が精力的に開発されている．Pelleg and 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，データのただ一度のスキャンだけでおおよそのクラスタリングを行い，それを改善するために付加的なスキャンを再度おこなうものである．

k-means 法には従来からも

- 種子点選定（初期分割）の違い
- 次の重心を計算するための手法の違い
- クラスター評価基準の改良
- データとクラスターとの関連づけに距離よりもむしろ確率密度を利用するといった違い

などによって幾つかのバリエーションがあり，特に種子点の選び方によって結果が変わってしまうという問題や，最適解に収束するとは限らないという問題（ISODATA 法は最適解を求めようとするアプローチであろうけれども）がある．しかしデータマイニングツールとしての見地からいえば，予めクラスター数を定めなければならないことは，より大きな制約であると考えられる．

Key words: clustering, k-means, BIC, information criterion, number of clusters

もちろんk-means法のクラスター数を種々に変えることによって，発見的に最適なクラスター数を求めるアプローチは可能である．事実，Hardy (1996) には，最適なクラスター数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）に対してさまざまな事例でもってその比較・評価を与えている．しかしどの方法を用いるにしても発見的に最適なクラスター数を求めることにかわりはなく，これらを用いる限りにおいては，せっかくのk-meansのもつ計算量の少なさのメリットが失われてしまう．

そこで本稿では，情報量規準の一つであるBIC (Bayesian Information Criterion; Schwarz (1978)) を用いることで，十分に小さなクラスター分割から始めて，各サブクラスターにおいて，分割が妥当と判断されるまで2分割を繰り返すアルゴリズムを支持し，これが有効に機能することを示す．また，その具体的な実装を提示する．このアイディア自体は既にPelleg and Moore (2000) にあるものであるが，以下の点が異なっている．

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ていること
2. 対数尤度の計算の一部に近似計算を用い，また2分割手続きに後述する実装上の工夫をして，計算速度の向上を図っていること
3. 最終的に生成されるクラスター数についての評価を与えていること

Pelleg and Moore (2000) ではサブクラスターの重心からの距離分散を一定と仮定している．逐次分割のネストが深くなると，分割の対象となるデータ数が少なくなり，それに伴い分散も一般に小さくなる; 分散の大きさの違いは考慮されるべきである．

また，プログラムの実装においても，本稿のそれは，逐次分割に関数の再帰呼び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，後で処理するようにしてある．これにより，逐次分割の階層が深くなったときに関数呼び出しのオーバヘッドを大幅に低減できる．

なお本プログラムは，統計言語Sのクローンとして知られるフリーソフトであるRで実装した．Rは2000年6月現在，バージョン1.1.0が公式版 (Official Release) として公開されており，性能および信頼性の点から十分な品質にあると評価できるものである．Rはhttp://www.r-project.org/から入手でき，LinuxやWindowsに対してはバイナリが提供されている．

2節ではk-means法の原理について述べ，3節に本稿の提案するアルゴリズムを示す．4節に最終的に選ばれたクラスター数についての評価を与え，5節をまとめとする．

## 2 k-means法

MacQueen (1967) によって提案されたk-means法は，

1. データ集合の中の最初の $k$ 個を1メンバのクラスターとして取り，
2. 残りのデータを最近隣距離の重心をもったクラスターに割り当て，

3. 現存するクラスターの重心を固定された種子点として取り，各データを最近隣距離の種子点に割り当てる操作をもう一度通過させるもの

である．しかし，実際に k-means と呼ばれる多くの（おそらくほとんど全ての）手法は，クラスターの重心が収束するまでデータの再配分を繰り返す．

多くのソフトウェアプログラムもそのように動作する．

# 3 x-means 法

x-means 法は本稿の基礎となるアイディアについて Pelleg and Moore (2000) が名づけたもので，k-means におけるクラスター数 $k$ が未知であることに由来する．x-means のアルゴリズムそのものはきわめて単純で，始めに十分に小さなクラスター分割から始めて，各サブクラスターについて 2 分割が適当であると判断される限り分割を繰り返すものである．

本稿で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 $\mathbf{x}_i$ に $p$ 変量正規分布

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

を仮定し，そのときの BIC を以下により計算する:

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

ここに $\widehat{\theta_i} = [\widehat{\mu_i}, \widehat{\mathbf{V}_i}]$ は，$p$ 変量正規分布の最尤推定値とする; $\mu_i$ は $p$ 次の平均値ベクトル，$\mathbf{V}_i$ 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，$\mathbf{V}_i$ 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．$\mathbf{x}_i$ 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．しかしながら後で述べるように，本稿で行ったシミュレーション結果からは，共分散を無視しても結果はあまり変わらなかった．

プログラムでは引数で指定 (ignore.covar=F に) すれば，共分散を考慮することができる．特に指定しなければ（デフォルトでは）ignore.covar=T となり，共分散は無視される．

6. $C_i^1, C_i^2$ のそれぞれに対して，パラメータ $\theta_i^1, \theta_i^2$ 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

$$\mathbf{x}_i \sim \alpha_i [f(\theta_i^1;\mathbf{x})]^{\delta_i}[f(\theta_i^2;\mathbf{x})]^{1-\delta_i} \tag{1}$$

とおく．ここで

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

とする．($\mathbf{x}_i$ は $C_i^1$ か $C_i^2$ のいずれか一方に必ず含まれる．) また $\alpha_i$ は，(1) 式を確率密度とするための基準化定数で，

$$\alpha_i = 1/\int [f(\theta_i^1;\mathbf{x}_i)]^{\delta_i}[f(\theta_i^2;\mathbf{x}_i)]^{1-\delta_i}\mathrm{d}\mathbf{x}$$

である ($1/2 \le \alpha_i \le 1$)．しかし厳密に $\alpha_i$ を求めようとすると $p$ 次積分が必要となり，多くの計算量を必要とする．ここでは $\alpha_i$ 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し，$\beta_i$ は $f(\theta_i^1;\mathbf{x}_i)$ と $f(\theta_i^2;\mathbf{x}_i)$ の分離の程度を示す指標で

$$\beta_i = \sqrt{\frac{\|\mu_1-\mu_2\|^2}{|\mathbf{V}_1|+|\mathbf{V}_2|}}$$

で示すものとする．$\beta_i = 0, 1, 2, 3$ のとき，$\alpha_i$ はそれぞれ $0.5/0.500 = 1$, $0.5/0.841 = 0.59$, $0.5/0.977 = 0.51$, $0.5/0.998 = 0.50$ となる．この近似は $f(\theta_i^1;\mathbf{x}_i)$ と $f(\theta_i^2;\mathbf{x}_i)$ の両方に正規分布を仮定したときに，両変量の差の分布が，平均が両分布の平均の差で，分散が両分布の分散の和で示される正規分布に従うことを利用したものである．

$\alpha_i$ が常に 1 ではないことは，$f(\theta_i^1;\mathbf{x}_i)$ と $f(\theta_i^2;\mathbf{x}_i)$ の分布が互いに大きく離れている場合（$\beta_i$ が大きい値の場合）を考えれば，

$$\int [f(\theta_i^1;\mathbf{x}_i)]^{\delta_i}[f(\theta_i^2;\mathbf{x}_i)]^{1-\delta_i} d\mathbf{x} \to 2$$

になることから容易に理解できる（このとき $\alpha_i = 1/2$ である）．(1) 式は，信頼性工学の分野では競合危険モデル (competing risks model) と呼ばれるモデルと同じ形をしており，この場合の定数項 $\alpha_i$ の大きさについては市田・鈴木 (1984) などに詳しい．

この 2 分割モデルにおける BIC を以下により計算する:

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

ここに $\widehat{\theta}_i' = [\widehat{\theta}_i^1, \widehat{\theta}_i^2]$ 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. $\mathrm{BIC} > \mathrm{BIC}'$ 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. $\mathrm{BIC} \leq \mathrm{BIC}'$ 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

これより本稿での適用については BIC が最適であると考えた．

# 4 性能評価

## 4.1 決定したクラスターの数に関する検討

(1) 以下の 2 変量正規乱数を各 50 個，計 250 個作成する．

$$x_j \sim N(\mu = [0, 0],\ \sigma = [0.2, 0.2]),\ (j = 1, \cdots, 50)$$

$$x_j \sim N(\mu = [-1,-1],\ \sigma = [0.2, 0.2]),\ (j = 51, \cdots, 100)$$
$$x_j \sim N(\mu = [1,1],\ \sigma = [0.2, 0.2]),\ (j = 101, \cdots, 150)$$
$$x_j \sim N(\mu = [2,2],\ \sigma = [0.2, 0.2]),\ (j = 151, \cdots, 200)$$
$$x_j \sim N(\mu = [3,3],\ \sigma = [0.2, 0.2]),\ (j = 201, \cdots, 250)$$

ここで $\mu$ は平均，$\sigma^2$ は分散を示す．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．2 変量正規乱数は，その都度，すなわち 1,000 組を発生させた．k-means のアルゴリズムは，R に実装されている Hartigan and Wong (1979) を用いた．

この（分割規準に BIC を適用した）x-means によって得られた最終的なクラスター数をまとめたのが表 1の上段である．1,000 回の繰返しのうち，正しいと考えられる 5 つのクラスターに分割されるのが最も多く 533 回である．次に多いのが 6 つのクラスターに分割される場合で 317 回である．4 つのクラスターに分割されるものは唯の一度もなかった．

中段には，分割規準に BIC ではなく AIC（赤池の情報量規準; Akaike's Information Criterion）を適用した場合の x-means の結果を示す．BIC を用いた場合に比べ，多めのクラスターに分割される傾向のあることがわかる．

下段には参考として，k-means においてクラスター $k$ をさまざまに変えたときに，最適なモデルと考えられる（AIC が最少となる）場合の $k$ を発見的に求めて，その回数を示した．これより分割規準に BIC を用いた x-means によって得られるクラスター数の分布が，発見的方法によって得られたクラスター数の分布とかなりよく一致していることがわかる．

表 1: クラスター中心が一直線上に並ぶ 2 変量正規乱数 250 個のクラスター分類

| クラスター数 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 ～ | 合計 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| x-means 法（BIC） | 0 | 533 | 317 | 108 | 34 | 8 | 0 | 0 | 0 | 0 | 1,000 |
| x-means 法（AIC） | 0 | 365 | 302 | 182 | 75 | 42 | 19 | 8 | 5 | 2 | 1,000 |
| 発見的方法 | 74 | 519 | 279 | 94 | 26 | 7 | 1 | 0 | 0 | 0 | 1,000 |

x-means では一度分割したクラスターを再び併合することをしない．このため x-means では，新た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ようである．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，したがって全体で 6 つのクラスターになる例が散見された．

(2) 次に以下の 2 変量正規乱数を各 50 個，計 250 個作成し，(1) と同様のシミュレーションを行った．

$$x_j \sim N(\mu = [0,0],\ \sigma = [0.2, 0.2]),\ (j = 1, \cdots, 50)$$
$$x_j \sim N(\mu = [-2,0],\ \sigma = [0.3, 0.3]),\ (j = 51, \cdots, 100)$$
$$x_j \sim N(\mu = [2,0],\ \sigma = [0.3, 0.3]),\ (j = 101, \cdots, 150)$$

$$x_j \sim N(\mu = [0, 2],\ \sigma = [0.4, 0.4]),\ (j = 151, \cdots, 200)$$
$$x_j \sim N(\mu = [0, -2],\ \sigma = [0.4, 0.4]),\ (j = 201, \cdots, 250)$$

(1) 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，(2) 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．この場合の結果を表 2に示す．

表 2: クラスター中心が十字に配置される 2 変量正規乱数 250 個のクラスター分類

| クラスター数 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 合計 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| x-means 法（BIC) | 2 | 6 | 9 | 469 | 383 | 99 | 27 | 5 | 0 | 0 | 0 | 0 | 0 | 1,000 |
| x-means 法（AIC) | 2 | 1 | 1 | 322 | 295 | 162 | 93 | 54 | 36 | 17 | 11 | 2 | 4 | 1,000 |
| 発見的方法 | 0 | 2 | 37 | 559 | 265 | 90 | 35 | 8 | 4 | 0 | 0 | 0 | 0 | 1,000 |

分割規準に BIC を適用した x-means によって得られたクラスター数の分布をみるに，発見的方法によって得られた最適クラスター数の分布とかなりよく適合していることがわかる．また，分割規準に AIC を適用した x-means は，BIC を適用した方法に比べ，多めのクラスター数が得られることがわかる．これらの事柄は (1) のシミュレーション結果と同じである．ただ，(2) のシミュレーションでは，x-means で 4 以下のクラスター数が得られる場合もわずかながら生じた．

(3) 共分散が 0 でない場合の例として，母相関係数 $r = 0.5$ の 2 変量正規乱数を各 50 個，計 250 個作成した．各 50 個の平均と分散の大きさは (2) の例と同じとする．シミュレーションで用いた乱数の一例を図 1に示す．また，この場合の結果を表 3に示す．

表 3: 母相関係数 $r = 0.5$ のクラスター 5 個より構成され，各クラスター中心が十字に配置される 2 変量正規乱数 250 個のクラスター分類

| クラスター数 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15–23 | 合計 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| x-means 法（BIC) | 13 | 9 | 1 | 345 | 388 | 166 | 61 | 12 | 4 | 1 | 0 | 0 | 0 | 0 | 1,000 |
| x-means 法（AIC) | 3 | 1 | 1 | 54 | 104 | 158 | 139 | 135 | 127 | 86 | 73 | 31 | 36 | 52 | 1,000 |
| 発見的方法 | 0 | 6 | 26 | 179 | 225 | 179 | 164 | 91 | 65 | 38 | 9 | 9 | 5 | 4 | 1,000 |

表 3の結果で目立つのは，発見的方法による結果の悪さである．このため分割規準に BIC を適用した x-means によって得られたクラスター数の分布と，発見的方法によって得られた最適クラスター数の分布とはあまり適合してはいない．しかしながら分割規準に BIC を適用した x-means の最終的に得られるクラスターの数については，表 2の結果に比べ，多めのクラスターが得られる事例が若干多くなったものの，概ね良好な結果が得られているように思われる．分割規準に AIC を適用した x-means の方が，BIC を適用した方法に比べ，多くのクラスターが得られるということについては，(1)(2) の

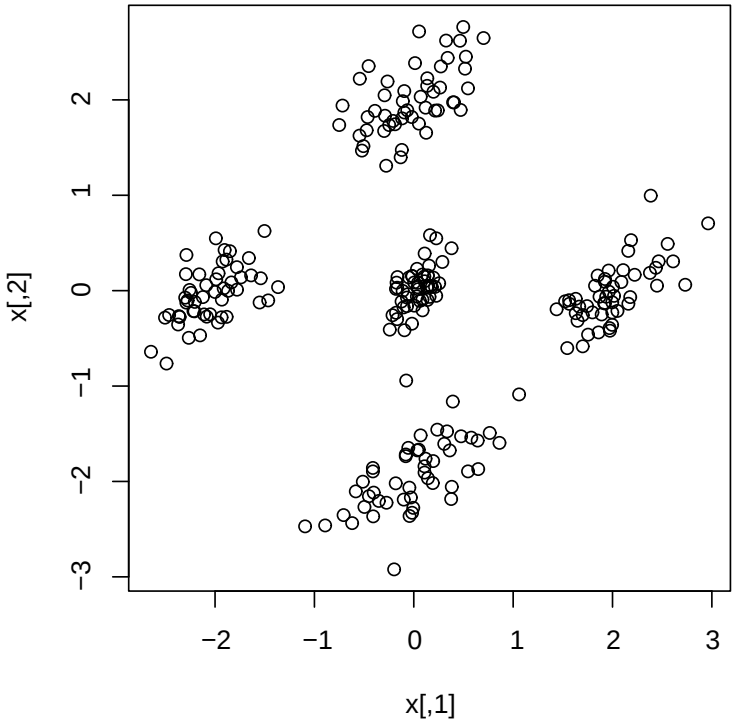


図 1: 母相関係数 $r = 0.5$ の 2 変量正規乱数 250 個の例

場合と同様である．

### 4.2　計算量に関する考察

k-means の計算量が $\mathcal{O}(kN)$ であることに注意すれば，x-means では（いくら小さなクラスター分割から始めて 2 分割を繰返したとしても）最終的に $k$ 個のクラスターを見つける必要があり，これを特定するために，すなわちその $k$ 個に対してさらに 2 分割をしてその分割が適当でないことを評価するために，結局 k-means の 2 倍の計算量が必要となることがわかる．もっとも BIC を計算するための計算量は必要だが，クラスター内の要素が確定した後に，$p$ 変量正規分布の平均と分散行列から一度だけ計算すればよく，(もちろん実装の仕方にはよるが）全体に比較すれば微小であろう．

一般に最適な $k$ を発見的に見つけるためには，

1. 評価関数が $k$ に対して単峰であることが仮定でき，かつ
2. 最適な $k$ に対して $k-1, k, k+1$ の 3 点による評価ができて

はじめて可能となるものである．すなわち，1. のきつい制約のもとで，評価関数の少なくとも 3 点による評価が必要である．x-means は，評価関数に相当する k-means による計算量の 2 倍の計算量で，特に制限を必要とせずに，(必ずしも最適点を選択するという保証はないものの）最適点に近い解を得ることができるわけで，計算コスト的に見合った成果を提示するといってよいであろう．

## 5　まとめ

k-means 法の逐次繰返しと BIC による分割停止基準を用いることで，情報理論的に最適と考えられるクラスター数を自動的に決定するアルゴリズムを提示し，そのプログラムを提示した．直感よりわずかに多めのクラスター数が選択される傾向のあるものの，先験情報が全くないときに，発見的な方法に拠らずに k-means のおおよそ 2 倍強の計算量で最適なクラスター数を求めることができる．

k-means については，近年より高速なアルゴリズムが提案されており，そのプログラムのソースコードが Fortran もしくは C で入手できれば，それらを容易に R や S に埋め込むことができる．したがって，x-means は，その高速な k-means を用いて利用できることを付記しておく．

**謝 辞**　本稿の不備をご指摘いただいた匿名の査読者，ならびに編集理事に厚くお礼申し上げます．

# 付録

# A　x-means のプログラム

## A.1　呼び出し形式

$p$ 変量のデータ行列 $\mathbf{x}$ ($n$ 行 $p$ 列) に対し，初期分割 $k_0 = 2$，k-means における重心計算のための繰返し数の最大値 20 として，x-means を

```
> xmeans(x, 2, 20)
```

によって計算すると，表 4に示す結果を得る．

出力は以下を意味する．

`$cluster` 各データが属するクラスターの番号

`$centers` 各クラスターの重心

`$lnL` $p$ 変量正規分布を仮定した場合の各クラスターの対数尤度

`$size` 各クラスターに含まれる要素数

表 4: xmeans(x,2,20) の出力

```
> xmeans(x,2,20)
$cluster
  [1] 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5
 [38] 5 5 5 5 5 5 5 5 5 5 5 5 5 2 2 5 2 2 2 2 2 2 2 2 5 2 2 2 2 2 2 2 2 2 2 2
 [75] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 5 5 2 2 2 2 3 3 3 3 3 3 3 3 3 3
[112] 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
[149] 3 3 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
[186] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4
[223] 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4

$centers
         [,1]        [,2]
1  3.00814345  2.98592860
2  0.98729531  0.99626808
3  1.95927156  1.98935766
4 -1.01638885 -1.01923197
5  0.05595856  0.05214385

$lnL
[1]  11.64117  29.36595  29.01711  20.61070 -12.06505

$size
[1] 50 46 50 50 54
```

## A.2 リスト

本プログラムのソースコードは，http://www.rd.dnc.ac.jp/~tunenori/xmeans.html より入手できる．

## 参 考 文 献


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

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

Hartigan, J.A. and 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.

市田嵩・鈴木和幸 (1984): 信頼性の分布と統計，信頼性工学シリーズ 3，日科技連出版社．

MacQueen, J.B. (1967): Some Methods for Classification and Analysis of Multivariate Observations, *Proceedings of the fifth Berkeley Symposium on Mathematical Statistics and Probability* **1**, University of California Press.

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

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

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

Vesanto, J. and Alhoniemi, E. (2000): Clustering of the Self-Organizing Map, *IEEE Transactions on Neural Networks* **11**(3), 586-600.

Vesanto, J. , Himberg, J. , Alhoniemi, E. and Parhankangas, J. (1999): Self-Organizing Map in Matlab: the SOM Toolbox, *Proceedings of the Matlab DSP Conference 1999*, Espoo, Finland, November, 35–40.

Yang, M-H. and Ahuja, N. (1999): A Data Partition Method for Parallel Self-Organizing Map, *Proceeding of the 1999 IEEE International Joint Conference on Neural Networks (IJCNN 99)*, Washington DC, July.

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

著者連絡先： 〒 153-8501　東京都目黒区駒場 2-19-23
大学入試センター　研究開発部　情報処理部門
Phone: 03-3468-3311　E-mail: tunenori@rd.dnc.ac.jp