情報処理学会論文誌 Vol. 50 No. 5 1493–1505 (May 2009)

ユークリッド距離の高速高精度推定と範囲問合せへの応用

城戸 健太郎[†1,*1] 桑島 洋[†1,*2] 鷲尾 隆[†1]

本稿では，ユークリッド距離行列（Euclidean Distance Matrix; EDM）内の限られた既知要素，すなわち限られた既知の事例間距離をもとに，それ以外の未知要素の推定値をその許容誤差幅とともに導出する新たな手法を提案する．さらに，この推定手法を適用した新たな効率的範囲問合せ手法を提案する．また，これらを既存手法と比較し，効率性と精度の両面から本提案手法が優れていることを示す[*1]．

Efficient and Accurate EDM Estimation and Its Application to Range Queries

Kentarou Kido,[†1,*1] Hiroshi Kuwajima[†1,*2] and Takashi Washio[†1]

This paper proposes a novel approach to estimate admissible values and their intervals of missing elements in an Euclidean Distance Matrix (EDM) based on limited known distance values among given objects in Euclidean space. Furthermore, this paper presents a new efficient range query approach by applying this estimation method. The superior performances of these approaches in both efficiency and accuracy are demonstrated through comparisons with some conventional approaches.

†1 大阪大学産業科学研究所第 1 研究部門（情報・量子科学系）知能推論研究分野
Department of Reasoning for Intelligence, The Institute for Scientific and Industrial Research, Osaka University
*1 現在，兼松株式会社
Presently with KANEMATSU CORPORATION
*2 現在，マイクロソフトディベロップメント株式会社
Presently with Microsoft Development Co., Ltd.
*1 この研究は，部分的に日本学術振興会科学研究費 19024048 および 19200013 を用いて行われた．

1. はじめに

近年，計算機の発達にともない，大量かつ大規模次元のデータを取り扱う機会が増大しており，それらの高速な処理の重要性が急速に高まってきた．特に，問合せ検索，クラスタリング，分類などの様々なデータ解析では，事例間の類似性計算が重要な基礎技術である．事例間の類似性は何らかの尺度で評価されるが，ベクトルで表される事例間の類似性を示す代表的尺度として，ユークリッド距離があげられる．本研究では事例間のユークリッド距離の高速推定と，それを用いた範囲問合せ検索を研究対象として取り上げる．

n 個の事例間のユークリッド距離を直接計算するにあたり，各事例間距離の計算コストを m とすると，全体の計算時間複雑性は $O(mn^2)$ となる．したがって，事例数や各事例を表すベクトル次元数の増加にともない，高速にすべての事例間の類似性距離を計算することは困難となる．たとえば，多くの文書検索や文書分類手法では，各文書に含まれる単語集合を数万次元の単語ベクトルで表し，膨大な文書から類似したベクトルで表される文書を高速に求める必要がある[29]．しかし，すべての文書間の類似性を検索・分類時やその前処理で計算することは非現実的である．同じく，化学物質相互の親和性評価実験や遺伝子発現実験などのように，多数の化学物質や発現遺伝子間の類似性距離を直接実験測定する必要がある場合にも，多事例間の類似性評価のために時間的，経済的に膨大なコストが必要となる．たとえば，生体遺伝子に突然変異を引き起こす可能性のある変異原生化学物質は 2 万種以上知られており，これらの分子や生体分子との相互作用をすべて実験的に確かめることは容易ではない[13]．また遺伝子発現実験では，マイクロアレーによって 1 度に 5 千個の遺伝子発現活性を測定可能であるが，実験の条件制御や測定誤差，コストなどの制約により，詳細な発現類似性を解析できる遺伝子組合せは限られる[19]．以上のように，計算量コストや実験制約から，事例間のすべての距離を直接計算・測定することは困難な場合が多い．

これらの問題を軽減する方法として，事例間の一部のユークリッド距離の計算値または測定値から，残りのすべての事例間のユークリッド距離を効率的に推定することが考えられる．これを実現するために利用可能な既存手法としては，潜在的意味インデクシング（Latent Semantic Indexing; LSI）[25]，行列補完[22]，半正定（Positive Semi-Definite; PSD）行列近似[3] などがあげられる．LSI では，特異値分解を用いて事例を表す m 次元ベクトルを k 次元空間（$k \leq m$）に射影し，事例間のユークリッド距離を近似することで，計算時間を抑えることができる．しかしながら，この手法は，各事例がベクトルデータとして与えられることを必要とし，一部事例間の距離データのみから残りすべての事例間距離を推定する

ものではない．さらに，近似精度の直接的制御が現状では困難である．行列補完[22]は，推定対象の行列の性質を利用して，既知の行列要素から未知の行列要素を，直接計算・測定することなく推定する．しかし，その計算時間複雑性が $O(n^6)$ にもなり，かつ十分な精度で推定するためには，ほぼ $O(n^2)$ の既知要素数を必要とする．また，半正定行列近似（以後，PSD 近似と呼ぶ）では，半正定性（positive semi-definite; PSD）を満たす類似性尺度について，一部の事例間の類似性尺度値から他の事例間の類似性尺度値を高速に近似推定する[3]．ただし，この手法の近似推定対象は PSD 行列に限られ，ユークリッド距離行列（Euclidean Distance Matrix; EDM）の推定には適用できない．また，推定精度の誤差区間幅を評価する方法も知られていない．

以上のように，各従来手法では，限られた既知の事例間距離，すなわちユークリッド距離行列（EDM）の限られた既知要素をもとに，それ以外の未知の事例間距離，すなわち EDM の未知要素を高速，高精度に推定することが困難である．そこで本稿では，我々が過去に提案した半正定行列推定手法（以後，PSD 推定と呼ぶ）[20),21)]と，ユークリッド距離と PSD 類似性尺度間の変換[12),27)]とを組み合わせることで，ユークリッド距離を推定する手法を提案する．前述のように，全事例間のユークリッド距離の直接計算ないし直接測定には $O(mn^2)$ のコストを要する．一方で，本提案手法では，ピボット事例と呼ぶ事例空間の基底事例を選択し，それらピボット事例とその他の事例間のユークリッド距離を用いて，すべての事例間の距離を推定する．ピボット事例は事例空間内で事例が分布する領域を近似的に包摂する部分空間を表現する基底事例であり，ピボット事例数がその部分空間の次元となる．ピボット事例数を k とすると，全事例間のユークリッド距離の推定処理コストは $O(kmn + kn^2)$ となる．これより，データを十分包摂するのに必要な部分空間の次元が低い，すなわち $k < m$ の場合は，直接処理に比べて推定処理の高速化が期待できる．本提案手法を，ここでは Euclidean distance matrix（EDM）推定と呼ぶ．

さらに本稿では，EDM 推定を用いたユークリッド距離による範囲問合せ手法を提案する．範囲問合せ（range query）とは，ユークリッド距離空間のような計量空間内で，ある事例に対して指定した閾値より近い類似したすべての事例を検索する問題である．これはきわめて基礎的な問題であり，その効率的な解法は多くの応用分野で必要とされる．たとえば，膨大な文書や化学物質のデータベースから，前述したように類似した文書や化学物質を高速に検索するニーズは大きく，少しでも高速な検索手法が求められている．計量空間における範囲問合せ手法は，その原理によって，ピボット事例に基づく手法とクラスタリングに基づく手法の 2 つに分類される．前者には，aesa，laesa[26)]，spaghettis[7)]，fg-trees[4)]，fg-arrays[9)] などがあげられる．laesa や spaghettis は，比較的大きな記憶容量を必要とするが，最も高速なアルゴリズムであることが知られている[12)]．また，後者には，M-trees[10)]，Voronoi trees[15)]，gh-trees[31)]，lists of clusters[8)]，gna-tree[6)] などがあげられる．特に，M-trees は，高次元データに対しても比較的高速に動作するアルゴリズムとして知られている[12)]．

しかしながら，一般的には範囲問合せに対するアルゴリズムの効率は，高次元データに対して低いことが知られている[12),27)]．この原因としては，単純に距離を計算するコストが空間次元 m に比例するためであること以外に，より本質的で重要な次元の呪いの問題があげられる．すべての次元に対して，事例が独立かつ同一な分布（$i.i.d.$）に従うデータ集合を仮定したとき，p_i，q_i をそれぞれ事例ベクトル p，q の i 番目の要素とすると，データ集合内の任意の 2 事例 p，q のユークリッド距離は $d(p,q) = \sqrt{\sum_{i=1}^{m}(p_i - q_i)^2}$ で与えられる．このとき，任意の事例 p，q について，$\frac{1}{m}\sum_{i=1}^{m}(p_i - q_i)^2$ は上述の（$i.i.d.$）分布の分散に近い．したがって，$d(p,q)$ はデータ集合内のすべての 2 事例間の平均距離に近く，次元数 m が大きいと，その分散は小さい．以上のことから，範囲問合せにおいて，計量空間の特徴である三角不等式を利用して効率的計算削減を実現することが困難となる．

この問題を軽減する手段として，本稿では EDM 推定を適用して，事例 p，q 間の正確な距離とその精密な誤差幅を効率的に推定することを考える．範囲問合せ問題を解くに際し，EDM 推定により得られたより正確な推定距離下限値を用い，問合せ事例との距離が一定閾値以下の事例を選択し，その距離のみを直接計算することで，計算量を効率的に削減する．

本研究の第 1 の目的は前述の EDM 推定原理とアルゴリズムの提案である．また，第 2 の目的は，高次元ユークリッド距離空間における新しい効率的範囲問合せアルゴリズムの提案である．本稿で提案する範囲問合せアルゴリズムを，matrix estimation based approximating and eliminating search algorithm（maesa）と呼ぶ．

2. 関 連 研 究

2.1 EDM と PSD 行列間の変換

n 個の全事例を表す整数の集合を $OB = \{1, \cdots, n\}$ としたとき，その 2 事例 p，$q \in OB$ 間のユークリッド距離の 2 乗 $d_{p,q}^2$ を第 p，q 要素とする $n \times n$ の行列 D^2 をユークリッド距離行列（EDM）という．過去のいくつかの行列補完研究において，EDM を補完するために，ユークリッド距離を PSD 類似性尺度に変換する式 (1) ないし式 (2) が用いられている[1),23)]．

$$a_{p',q'} = \frac{d_{p,1}^2 + d_{q,1}^2 - d_{p,q}^2}{2d_{p,1}d_{q,1}} \quad (p,q>1), \tag{1}$$

ここで，$p' = p-1$，$q' = q-1$ であり，$a_{p',q'}$ は $(n-1)\times(n-1)$ の PSD 行列 A の p'，q' 要素である．

$$a_{p,q} = \exp\left(-\lambda d_{p,q}^2\right) \tag{2}$$

ここで，パラメータ λ は正の実数であり，A は $n\times n$ の PSD 行列である．また，いずれの変換においても A の各対角要素は 1 となる．前者は 2 つの数値ベクトルのユークリッド距離（Euclidean distance）と相関（Correlation）の関係を表すよく知られた式である．後者は Schoenberg transform と呼ばれるガウスカーネル（Gauss Kernel）の一例である．本稿では略して，前者の式 (1) を *EC-transform*，後者の式 (2) を *SG-transform* と呼ぶ．

2.2　PSD 近似

PSD 近似は PSD 類似性尺度行列 A 中で，限られた既知の行列要素値を用いて，他の要素値を効率的に近似計算する手法である[3),16)]．PSD 近似では，不完全コレスキー分解に基づく漸化的な因数分解によって近似を行う．この手法は，類似性尺度行列の大部分が急速に減衰するスペクトルを持つ，すなわち，階数が低い場合に良い近似を与えることが知られている[32)]．n 個の全事例集合 OB 内の k（$\leq n$）個の事例からなる部分集合を $OA^k \subseteq OB$（$k = \left|OA^k\right|$）とする．ここで，A^k を $k\times k$ の A の主小行列とし，OA^k 内の k 個の事例間の類似性尺度を表すものとする．さらに，B^{n-k} を $k\times(n-k)$ の A の部分行列とする．B^{n-k} は OA^k 内のすべての事例と $OB^{n-k} = OB - OA^k$ 内のすべての事例との間の類似性尺度で構成される．さらに OB^{n-k} 内の事例どうしの類似性尺度を表す $(n-k)\times(n-k)$ の行列を X^{n-k} とする．これらの定義から，以下の $A(k)$ を考える．

$$A(k) = \begin{pmatrix} A^k & B^{n-k} \\ {B^{n-k}}^T & X^{n-k} \end{pmatrix} \tag{3}$$

PSD 近似はトレース $tr(A-A(k))$ をある許容範囲内にするように A^k や B^{n-k} から行列 X^{n-k} の各要素を近似的に導く手法である．この近似は次に示すように，$A(k)$ に k ステップ不完全コレスキー分解を適用することにより行われる．

$$A(k) = L(k)U(k) \tag{4}$$

ただし，

$$U(k) = \left(U^k U^{n-k}\right) = \left(\begin{array}{ccc:ccc} u_{1,1} & \cdots & u_{1,k} & u_{1,k+1} & \cdots & u_{1,n} \\ & \ddots & & \vdots & & \vdots \\ 0 & & u_{k,k} & u_{k,k+1} & \cdots & u_{k,n} \end{array}\right)$$

$$L(k) = \begin{pmatrix} L^k \\ L^{n-k} \end{pmatrix} = \left(\begin{array}{ccccc} 1 & & & & 0 \\ l_{2,1} & \ddots & & & \\ \vdots & \ddots & 1 & & \\ l_{k,1} & \cdots & l_{k,k-1} & 1 & \\ \hdashline l_{k+1,1} & \cdots & l_{k+1,k-1} & l_{k+1,k} & \\ \vdots & & \vdots & \vdots & \\ l_{n,1} & \cdots & l_{n,k-1} & l_{n,k} & \end{array}\right)$$

$$u_{1,j} = a_{1,j}, \quad l_{j,i} = \frac{u_{i,j}}{u_{i,i}}$$

$$u_{i,j} = a_{i,j} - \sum_{h=1}^{i-1} \frac{u_{h,i}}{u_{h,h}} u_{h,j} \tag{5}$$

であり，$a_{i,j}$ は $A(k)$ の i，j 要素である．さらに，以下に示すように，$k+1 \leq p \leq n$ における $U(k)$ 内の各列の残差を定義する．

$$e_p^{(k)} \equiv u_{p,p} = 1 - \sum_{h=1}^{k} \frac{u_{h,p}^2}{u_{h,h}} \tag{6}$$

$A(k)$ の近似誤差であるトレース $tr(A-A(k))$ は $p = (k+1),\ldots,n$ の残差の総和となる．

$k=1$ から開始し，行列の上部から下部に向かって $U(k)$ を構成する．$k+1$ 段階では，$tr(A-A(k))$ をより小さくするように，ピボット事例 p を OB^{n-k} から選択し，OA^k へ移動する．この操作によって，OA^{k+1} と OB^{n-k-1} に更新され，$U(k)$ は以下の $(k+1)\times n$ 行列に拡張される．

$$U'(k+1) = \begin{pmatrix} U^k & u_p^k & {U'_{\bar{p}}}^{n-k-1} \\ 0 & 1 & {a_p^{n-k-1}}^T \end{pmatrix}$$

ただし，$U'^{n-k-1}_{\bar{p}}$ は U^{n-k} から u_p^k を取り除いた $k \times (n-k-1)$ の行列であり，${a_p^{n-k-1}}^T$ は，移動したピボット事例 $p \in OA^{k+1}$ と各事例 $q \in OB^{n-k-1}$ 間の類似性尺度からなる $n-k-1$ 次元ベクトルである．次に，$U'(k+1)$ に対して $k+1$ 段階の不完全コレスキー分解を以下のように適用する．

$$\begin{aligned} U(k+1) &= \left(U^{k+1} U^{n-k-1}\right) \\ &= \left(\begin{pmatrix} U^k & u_p^k \\ 0 & e_p^{(k)} \end{pmatrix} U_{\bar{p}}^{n-k-1} \right) \end{aligned}$$

ただし，$U_{\bar{p}}^{n-k-1}$ は各事例 $q \in OB^{n-k-1}$ に対応するベクトル $u_q^{k+1} = \begin{pmatrix} {u_q^k}^T & u_{k+1,q} \end{pmatrix}^T$ からなる $(k+1) \times (n-k-1)$ 行列である．$e_p^{(k)}$ は式 (6) によって計算され，$U_{\bar{p}}^{n-k-1}$ は式 (5) に基づき各事例 q に対して u_q^{k+1} 内の最後の要素 $u_{k+1,q}$ を計算することにより求まる．$A(k+1) = L(k+1)U(k+1)$ の導出後，より小さな $tr(A - A(k+1))$ が得られ，A の階数が低ければ，この漸化的な近似は早期に終了する．

2.3 PSD 推定

我々が過去に提案した PSD 推定[20),21)] は，PSD 近似と多くの原理を共有するが，$OA \subseteq OB$ 内のピボット事例と他事例の PSD 類似性尺度値を利用することで，各事例 $q \in OB$ と他のすべての事例 $p \in OB$ 間の類似性尺度値を “許容誤差範囲” を保証して導く点が異なっている．我々の PSD 推定が用いる重要な PSD 行列の性質として，以下に示す定理があげられる[5)]．

Theorem 1（シルベスターの判定法） 正方行列 S が半正定（$S \succeq 0$）であるための必要十分条件は，『行列 S のすべての主小行列式が 0 以上なこと』である．また，正方行列 S が正定（$S \succ 0$）であるための必要十分条件は，『行列 S のすべての主座小行列式が 0 より大きいこと』である． ■

この定理に基づき，我々の PSD 推定では行列 A の全要素の推定に際し，A のすべての主小行列式が非負であるという条件を用いる．ここで，$A(k)$ の主小行列である $(k+2) \times (k+2)$ の $A_{p,q}^{k+2}$ を考える．ただし，$A_{p,q}^{k+2}$ は以下のように，式 (3) における OA 内のピボット間の類似性尺度を表す $k \times k$ の主小行列 A^k，OB^{n-k} 内の事例 p および q と各ピボット間の類似性尺度を表す k 次元ベクトル a_p^k，$a_q^k \in B^{n-k}$，p，$q \in OB$ の類似性尺度を表す未知要素 $x_{p,q} (= x_{q,p}) \in X^{n-k}$ を含む．

$$A_{p,q}^{k+2} = \begin{pmatrix} A^k & a_p^k & a_q^k \\ {a_p^k}^T & 1 & x_{p,q} \\ {a_q^k}^T & x_{q,p} & 1 \end{pmatrix} \tag{7}$$

Theorem 1 より，$A \succeq 0$ から $\det\left(A_{p,q}^{k+2}\right) \geq 0$ かつ $\det\left(A^k\right) \geq 0$ が成り立つ．そして，本稿ではさらに強い制約である $\det\left(A^k\right) > 0$ を仮定する．これは後述の方法により，一般性を失わずに成立させることが可能である．これらの制約は以下に示す一般的な関係式によって，$x_{p,q}$ に関する不等式に変換可能である[2)]．

$$\det(A_{p,q}^{k+2}) = \det(A^k) \times \det\left(\begin{pmatrix} 1 & x_{p,q} \\ x_{q,p} & 1 \end{pmatrix} - (\boldsymbol{a}_p^k\ \boldsymbol{a}_q^k)^T (A^k)^{-1} (\boldsymbol{a}_p^k\ \boldsymbol{a}_q^k) \right)$$

ここで，$\det\left(A_{p,q}^{k+2}\right) \geq 0$ と $\det\left(A^k\right) > 0$ の仮定より，上記の式から以下に示す $x_{p,q}$ の 2 次不等式が得られる．

$$\det\left(\begin{pmatrix} 1 & x_{p,q} \\ x_{q,p} & 1 \end{pmatrix} - (\boldsymbol{a}_p^k\ \boldsymbol{a}_q^k)^T (A^k)^{-1} (\boldsymbol{a}_p^k\ \boldsymbol{a}_q^k) \right) \geq 0 \tag{8}$$

我々の PSD 推定では，この不等式を解くことで未知の要素 $x_{p,q} \in A$ の存在領域を決定する．なお，式 (8) は $A_{p,q}^{k+2}$ における局所的な制約であるため，最大の $det(A)$ は保証されない．それゆえ，我々の PSD 推定は PSD 補完とは異なる．

我々の PSD 推定では，実際には式 (8) を直接解いて解を得る代わりに，その制約を各事例の因数に分解し，精度を保証しつつ効率的に各未知要素を計算するアルゴリズムを提案した[20),21)]．以下に我々の提案で導いたくつかの数学的定義と関連する定理，補題を通じて，そのアルゴリズムを説明する．まず，式 (7) の $A_{p,q}^{k+2}$ に前述した k ステップ不完全コレスキー分解を適用すると以下の式を得る．

$$A_{p,q}^{k+2} = {L'_{p,q}}^{k+2} {U'_{p,q}}^{k+2} \tag{9}$$

$$U'_{p,q}{}^{k+2} = \begin{pmatrix} U^k & u_p^k & u_q^k \\ 0 & 1 & x_{p,q} \\ 0 & x_{q,p} & 1 \end{pmatrix}$$

$$L'_{p,q}{}^{k+2} = \begin{pmatrix} L^k & 0 & 0 \\ l_p^{kT} & 1+\frac{l_p^{kT}\left(u_q^k x_{p,q}-u_p^k\right)}{1-x_{p,q}^2} & \frac{l_p^{kT}\left(u_p^k x_{p,q}-u_q^k\right)}{1-x_{p,q}^2} \\ l_q^{kT} & \frac{l_q^{kT}\left(u_q^k x_{p,q}-u_p^k\right)}{1-x_{p,q}^2} & 1+\frac{l_q^{kT}\left(u_p^k x_{p,q}-u_q^k\right)}{1-x_{p,q}^2} \end{pmatrix}$$

この分解の妥当性は，式 (9) に $U'_{p,q}{}^{k+2}$ と $L'_{p,q}{}^{k+2}$ を代入することで容易に確かめることができる．さらに以下に示す $v_p^{(k)}$ と $v_q^{(k)}$ を定義し，式 (6) とこれらの新しいベクトルにより残差 $e_p^{(k)}$，$e_q^{(k)}$ を表す．

$$v_p^{(k)} = \left(\begin{array}{cccc} u_{1,p}/\sqrt{u_{1,1}} & u_{2,p}/\sqrt{u_{2,2}} & \cdots & u_{k,p}/\sqrt{u_{k,k}} \end{array} \right)^T \tag{10}$$

$$v_q^{(k)} = \left(\begin{array}{cccc} u_{1,q}/\sqrt{u_{1,1}} & u_{2,q}/\sqrt{u_{2,2}} & \cdots & u_{k,q}/\sqrt{u_{k,k}} \end{array} \right)^T \tag{11}$$

$$e_p^{(k)} = 1 - \left|v_p^{(k)}\right|^2, \quad e_q^{(k)} = 1 - \left|v_q^{(k)}\right|^2$$

このとき，式 (8) の解 $x_{p,q}$ は以下の定理により導かれることが分かっている[20),21)]．

Theorem 2　$A_{p,q}^{k+2}$ における未知要素の存在領域，すなわち式 (8) の解は以下で与えられる．

$$\hat{x}_{p,q}^{(k)} - \Delta x_{p,q}^{(k)} \le x_{p,q} \le \hat{x}_{p,q}^{(k)} + \Delta x_{p,q}^{(k)}$$

ただし，

$$\hat{x}_{p,q}^{(k)} = v_p^{(k)T} v_q^{(k)}, \quad \Delta x_{p,q}^{(k)} = \sqrt{e_p^{(k)}}\sqrt{e_q^{(k)}}$$ ■

これにより，類似性尺度の推定値やその誤差領域が個々の事例 p ごとに因数分解形式で表されることが分かる．この性質は後述の範囲問合せに都合が良い．なぜならば，1 度 $p \in OB$ に関する $v_p^{(k)}$ や $\sqrt{e_p^{(k)}}$ を計算しておけば，必要なときに $\hat{x}_{p,q}^{(k)}$ や $\Delta x_{p,q}^{(k)}$ を簡単に計算できるからである．さらに，PSD 近似とは異なり，誤差を別途調べなくても，誤差領域 $\Delta x_{p,q}^{(k)}$ が A から得られることも利点である．

Theorem 2 より因数分解された解や誤差領域は，各事例 p の $u_p^{(k)}$ を知れば導くことができる．それには $A_{p,q}^{k+2}$ の上記 k ステップ不完全コレスキー分解ではなく，以下に示す $A_{p,q}^{k+2}$ の $(k+1)\times(k+1)$ 主座小行列の修正コレスキー分解を行えば十分である．

$$A_p^{k+1} = \begin{pmatrix} A^k & a_p^k \\ a_p^{kT} & 1 \end{pmatrix} \tag{12}$$

式 (3) の $A(k)$ は PSD 行列であるため，Theorem 1 よりその主小行列である A_p^{k+1} は正定（$A_p^{k+1} \succeq 0$）である．そのため，式 (5)，式 (6) を用いて，以下のように A_p^{k+1} を修正コレスキー分解することができる．

$$A_p^{k+1} = L_p^{k+1} U_p^{k+1}$$

ただし，

$$U_p^{k+1} = \begin{pmatrix} U^k & u_p^k \\ 0 & e_p^{(k)} \end{pmatrix}, \quad L_p^{k+1} = \begin{pmatrix} L^k & 0 \\ l_p^{kT} & 1 \end{pmatrix}$$

$$u_p^k = \left(\begin{array}{cccc} u_{1,p} & u_{2,p} & \cdots & u_{k,p} \end{array} \right)^T$$

$$l_p^k = \left(\begin{array}{cccc} u_{1,p}/u_{1,1} & u_{2,p}/u_{2,2} & \cdots & u_{k,p}/u_{k,k} \end{array} \right)^T$$

である．このとき，残差 $e_p^{(k)}$ に関して以下の補題が成り立つことが分かっている[20),21)]．

Lemma 1　式 (12) の A_p^{k+1} が半正定（$A_p^{k+1} \succeq 0$）かつその $k\times k$ 主座小行列 A^k が正定（$A^k \succ 0$）ならば，式 (6) の残差 $e_p^{(k)}$ は $0 \le e_p^{(k)} \le 1$ を満たす．さらにこのとき，$e_p^{(k)} > 0$ であるならば，A_p^{k+1} は正定（$A_p^{k+1} \succ 0$）となる． ■

この補題によって，以下に述べるように我々の PSD 推定の可能性がつねに保証される．$A(k)$ の不完全コレスキー分解を進めるあたり，PSD 近似と同様に次のピボット事例として，最大残差 $e_{\max}^{(k)} = \max_{p\in OB^{n-k}} e_p^{(k)}$ を持つ事例 $p_{\max} \in OB^{n-k}$ を選ぶ．$A^k \succ 0$ ならば，$e_{\max}^{(k)} > 0$ と Lemma 1 より $A^{k+1} = A_{p_{\max}}^{k+1} \succ 0$ となり，$A^1 = [1] \succ 0$ から漸化的に任意の k について $\det(A^k) > 0$ が保証される．これゆえ，本節のはじめに述べた仮定が成立し，つねに式 (8) の不等式に基づく PSD 推定が可能となる．

ここで，我々が許容する残差閾値 ε_{tol}（$0 < \varepsilon_{tol} < 1$）を与えたとき，事例 $p \in OB^{n-k}$ の残差 $e_p^{(k)}$ と前述の k ステップでの最大残差 $e_{\max}^{(k)} = \max_{p\in OB^{n-k}} e_p^{(k)}$ より，事例 p と他の事例 $q \in OB^{n-k}$ との距離の最大誤差領域幅が ϵ_{tol} 以下になる条件は，Theorem 2 より以下で与えられる．

$$\sqrt{e_p^{(k)}}\sqrt{e_{\max}^{(k)}} \le \varepsilon_{tol} \tag{13}$$

この条件を満たす事例 p に関しては，他の事例との類似性尺度を十分な精度で推定できたと考えられる．以上より，式 (13) を事例 p に関する類似性尺度推定の終了条件とする．

3. 提案する EDM 推定

本稿で提案する EDM 推定は 2 章で述べた変換法を用いて，PSD 推定を EDM 推定に拡張したものであり，EDM 推定は PSD 推定と同様に，精密な誤差領域とともに事例 p, $q \in OB$ 間の距離を推定する[18]．すなわち，以下の $n \times n$ の EDM $D^2(k)$ が与えられたとき，

$$D^2(k) = \begin{pmatrix} D^k & E^{n-k} \\ {E^{n-k}}^T & Y^{n-k} \end{pmatrix} \tag{14}$$

OB^{n-k} 内の事例間の距離の 2 乗に対応する $(n-k) \times (n-k)$ 行列 Y^{n-k} の各要素を，精密な誤差領域とともに，D^k や E^{n-k} から推定する．ただし，$k \times k$ 行列 D^k は OA^k 内のピボット事例間の距離の 2 乗を表し，$k \times (n-k)$ 行列 E^{n-k} は OA^k 内の各ピボット事例と OB^{n-k} 内の各事例間の距離の 2 乗を表す．

この問題は式 (1)，式 (2) の *EC-transform* と *SG-transform* を用いて，式 (14) を式 (3) の PSD 行列に変換し，PSD 推定によって解くことができる．これにより，因数分解された PSD 類似性尺度 $v_p^{(k)}$ の推定値やその誤差領域 $\sqrt{e_p^{(k)}}$ をすべての事例 $p \in OB$ に対して得ることができる．また，k 段階でのこの情報を記憶しておくことで，任意の 2 事例 p, $q \in OB$ に対する類似性尺度の推定値や誤差領域 $\Delta x_{p,q}^{(k)}$ を容易に求めることができる．変換の選択に関し重要なことは，EDM 推定においても PSD 推定同様に効率的かつ高精度な計算ができるように，推定値の誤差領域が PSD 領域だけでなく EDM 領域においても因数分解可能なことである．因数分解できれば，EDM 領域でも式 (13) と同様な終了条件を導入し，高速に高精度な計算が可能である．

EC-transform（式 (1)）と *SG-transform*（式 (2)）において，ユークリッド距離や PSD 類似性尺度で表現される推定値や誤差領域の関係は，それぞれ以下のようになる．

$$\hat{x}_{p,q}^{(k)} \pm \Delta x_{p,q}^{(k)} = \frac{d_{p,1}^2 + d_{q,1}^2 - \left(\hat{y}_{p,q}^{(k)} \mp \Delta y_{p,q}^{(k)}\right)}{2d_{p,1}d_{q,1}}$$

$$\hat{x}_{p,q}^{(k)} \pm \Delta x_{p,q}^{(k)} = \exp\left[-\lambda\left(\hat{y}_{p,q}^{(k)} \mp \Delta y_{p,q}^{(k)}\right)\right]$$

ただし，$\hat{y}_{p,q}^{(k)}$ と $\Delta y_{p,q}^{(k)}$ はそれぞれ $d_{p,q}^2$ の推定値と誤差領域を表す．また，$\Delta x_{p,q}^{(k)} = \sqrt{e_p^{(k)}}\sqrt{e_q^{(k)}}$ を用いると，これらの逆変換はそれぞれ以下のようになる．

$$\hat{y}_{p,q}^{(k)} \pm \Delta y_{p,q}^{(k)} = d_{p,1}^2 + d_{q,1}^2 - 2d_{p,1}d_{q,1}\hat{x}_{p,q}^{(k)} \pm 2d_{p,1}d_{q,1}\sqrt{e_p^{(k)}}\sqrt{e_q^{(k)}}$$

$$\hat{y}_{p,q}^{(k)} \pm \Delta y_{p,q}^{(k)} = -\frac{\log\left(\hat{x}_{p,q}^{(k)} \mp \sqrt{e_p^{(k)}}\sqrt{e_q^{(k)}}\right)}{\lambda}$$

ここで，前者の *EC-transform* では，EDM 領域で誤差領域を，厳密に以下のように因数分解できる．

$$\delta y_p^{(k)} = \sqrt{2}d_{p,1}\sqrt{e_p^{(k)}} \qquad (EC)$$

ただし，$\Delta y_{p,q}^{(k)} = \delta y_p^{(k)}\delta y_q^{(k)}$ である．しかしながら，後者はそれが困難である．そこで，後者の *SG-transform* を使用するために以下に示す近似を用いて因数分解を行う．

$$\delta y_p^{(k)} = \sqrt{\frac{e_p^{(k)}}{\lambda \exp\left[-\lambda\bar{y}\right]}} \qquad (SG)$$

これは以下に示すテイラー展開と平均値近似により，得られる．$\bar{y}$ は OB 内の事例間の距離の 2 乗の平均値である．

$$\begin{aligned}\hat{x}_{p,q}^{(k)} \pm \Delta x_{p,q}^{(k)} &= \exp\left[-\lambda\hat{y}_{p,q}^{(k)}\right]\exp\left[\pm\lambda\Delta y_{p,q}^{(k)}\right] \\ &\approx \exp\left[-\lambda\hat{y}_{p,q}^{(k)}\right] \pm \lambda\exp\left[-\lambda\hat{y}_{p,q}^{(k)}\right]\Delta y_{p,q}^{(k)} \\ &\approx \exp\left[-\lambda\hat{y}_{p,q}^{(k)}\right] \pm \lambda\exp\left[-\lambda\bar{y}\right]\Delta y_{p,q}^{(k)}\end{aligned}$$

これらの結果によって，ユークリッド距離空間における許容誤差 ε_{tol}^d が与えられたとき，それぞれ以下の式により，誤差幅を ε_{tol}^d 以下に制約する．

$$2d_{p,1}\sqrt{e_p^{(k)}}\sqrt{d^2 e_{\max}^{(k)}} \le \varepsilon_{tol}^d \qquad (EC) \tag{15}$$

$$\frac{\sqrt{e_p^{(k)}}\sqrt{e_{\max}^{(k)}}}{\lambda\exp\left[-\lambda\bar{y}\right]} \le \varepsilon_{tol}^d \qquad (SG) \tag{16}$$

ただし，$d^2 e^{(k)}_{\max} = \max_{p \in OB^{n-k}} d^2_{p,1} e^{(k)}_p$ である．EDM 推定では，式 (15)，式 (16) を推定の終了条件とする．

以上より，*EC-transform*（式 (1)）には 2 つの利点がある．第 1 に PSD 領域，EDM 領域の両方において，誤差領域を厳密に因数分解できることである．第 2 に余分なパラメータを含まないため，あいまいさがない．一方で，*SG-transform*（式 (2)）では，誤差領域を厳密には因数分解できず，しかも変換類似性尺度値はパラメータ λ の値に強く依存する．たとえば，$\lambda \cong +0$ ないし $\lambda \cong \infty$ であるならば，行列 A のほとんどすべての要素が，それぞれ 1 か 0 になる．これらの場合，D^2 のランクすら保存されない．*SG-transform* は，近年，統計数理やデータマイニングで多用されているガウスカーネル関数の一種であるが，EDM 推定では，それよりも *EC-transform* の方が有利な点が多いと考えられる．

4. maesa のアルゴリズム

4.1 事前構成アルゴリズム

範囲問合せ手法は，一般に高速な検索を可能にするために，検索対象データから種々の準備情報を構成し，実際の範囲問合せ時にそのデータと検索対象データから，指定された範囲内で問合せ事例に近い全事例を効率的に検索する．maesa の事前構成アルゴリズムを**図 1** に示す[18]．このアルゴリズムでは，$k = 1$ から始まり，検索対象データ集合 OB が空になるまで，以下に述べる 4 段階からなる反復処理を行う．

approximating：前回の反復処理で最大残差を持つ事例をピボット事例として選択し，それを OB から除外する．

distance computing：除外したピボット事例と OB 内の各事例間の距離を直接計算し，その結果を，不完全コレスキー分解の上三角行列である式 (4) の $U(k)$ にあたる配列 $U[k]$ の最右列に加える．そして，そのピボット事例をそれまでの反復で求めたピボット事例の順序付きリスト OA に加える．

updating lower bound：配列 $U[k]$ の 1 ステップ不完全コレスキー分解を行い，$U[k+1]$ と新たな残差ベクトル $E[k+1]$ を生成する．これらは因数分解された推定値や誤差幅を表す．

eliminating：式 (15) ないし (16) により，許容誤差幅以内で推定できた事例を OB から除外する．

各 k ステップの $E[k]$ および $U[k]$ は，後の範囲問合せアルゴリズムで使用するため記録する．

Pre-construction of Distance Factor Information
Input: $OB \subseteq OU$; *a dataset of objects*
$\epsilon^d_{tol} \in \mathcal{R}$; *Euclidean error tolerance*
Output: $K \in \mathcal{I}$; *the maximum steps*
$OA \in OB$; *an ordered list of pivots*
$OC^k \in OB$; *a completed set for each k*
$U[k] \in \mathcal{R}^{|OA| \times (|OA|+|OC^k|)}$; *factorized vectors for each k*
$E[k] \in \mathcal{R}^{|OA|+|OC^k|}$; *factorized error bounds for each k*
Function: $d^2 : OU \times OU \to \mathcal{R}$; *squared distance*
$EC : \mathcal{R} \to \mathcal{R}$; *EC-transform*
$EC^{-1} : \mathcal{R} \to \mathcal{R}$; *inversed EC-transform*
begin
  $k := 0$, $OA := \phi$, $U := [\,]$; *initializing*
  while $OB \neq \phi$ **do**
    $k := k + 1$;
    $pv := p_{max} \in OB$, $OB := OB - \{pv\}$; *approximating*
    $U := \begin{bmatrix} -\frac{U(OA,OA)}{0} & \frac{U(OA,\{pv\})}{1} & \frac{U(OA,OB)}{\{EC(d^2(pv,q))|q \in OB\}} \end{bmatrix}$;
    $OA(k) := \{pv\}$; *distance computing*
    $[U, E] := ICD(U)$; *one step incomplete Cholesky decomposition (updating lower bound)*
    $OC^k := CR(E, EC^{-1}, \epsilon^d_{tol})$; *column reduction*
    $OB := OB - OC^k$; *(eliminating)*
    $U[k] = [U(OA, OA)\ U(OA, OC^k)]$;
    $E(k) := [E(OA)\ E(OC^k)]$;
    $U = [U(OA, OA)\ U(OA, OB)]$;
  endwhile
  $K = k$;
end

図 1　maesa の事前構成アルゴリズム
Fig. 1　Pre-construction algorithm of maesa.

図 1 より，maesa の事前構成アルゴリズムのほぼすべての処理は while ループに含まれていることが分かる．その中で，最も時間を要する段階は，$O(m)$ の計算時間複雑性を有する距離計算を $|OB|(\leq n)$ 回行う distance computing である．さらに，$O(kn)$ の不完全コレスキー分解の 1 ステップや $O(n)$ の式 (15) ないし (16) の推定終了条件判定も主要な計算時間を要する．この反復の複雑性は $O(kn + mn)$ となる．OB は式 (15) ないし (16) の推定終了条件により，少ないステップ数 k（$\ll n$）で空になり，事前構成アルゴリズム全体の計算時間複雑性は $O(k^2 n + kmn)$ となる．一般的に EDM のランクはベクトル空間次元 m よりも小さいので $k < m$ となり，最終的に $O(kmn)$ となる．記憶容量複雑性は，ほぼ $U(k)$ を保存する記憶容量に左右される．式 (15) ないし (16) の推定終了条件により，

Range Query of $(q, r)_d$
Input: $q \in OU$; *a query object*
$r \in \mathcal{R}$; *a range for query*
$\epsilon^d_{tol} \in \mathcal{R}$; *Euclidean error tolerance*
$K \in \mathcal{I}$; *the maximum steps*
$OA \in OB$; *an ordered list of pivots*
$OC^k \in OB$; *a completed set for each k*
$U[k] \in \mathcal{R}^{|OA|\times(|OA|+|OC^k|)}$; *factorized vectors for each k*
$E[k] \in \mathcal{R}^{|OA|+|OC^k|}$; *factorized error bounds for each k*
Output: $Q \in OB$; *query result*
Function: $d^2 : OU \times OU \to \mathcal{R}$; *squared distance*
$EC : \mathcal{R} \to \mathcal{R}$; *EC-transform*
$EC^{-1} : \mathcal{R} \to \mathcal{R}$; *inversed EC-transform*
$lb : \mathcal{R}^{2k+2} \to \mathcal{R}$; *lower bound of similarity*
begin
$\boldsymbol{u}_q := [\,]$, $Q = \phi$; *initializing*
for every $k = 1 : K$ **do**
$a_{pv,q} = EC(d^2(OA(k), q))$; *distance computing*
$\boldsymbol{u}_q := \left(-\frac{\boldsymbol{u}_q}{a_{pv,q}} - \right)$;
if $a_{pv,q} \le r$ **then** $Q = Q + OA(k)$ **endif**
$[\boldsymbol{u}_q, e_q^{(k)}] := ICD^*([U[k]\ \boldsymbol{u}_q])$; *one step and last column incomplete Cholesky decomposition (updating lower bound)*
$Q = Q +$ *updating query result*
$\{p \in OC^k | \sqrt{EC^{-1}(lb(\boldsymbol{u}_p, \boldsymbol{u}_q, e_p^{(k)}, e_q^{(k)}))} \le r$ *where*
$\boldsymbol{u}_p := U[k](OA(1:k), \{p\}), e_p^{(k)} := E[k](\{p\})\}$;
endfor
compute $d^2(q, p)$ *for all* $p \in Q$; *distance computing*
$Q = \{p \in Q | \sqrt{(d^2(p, q))} \le r\}$; *finalizing query result*
end

図 2　*maesa* の範囲問合せアルゴリズム
Fig. 2　Range query algorithm of *maesa*.

$\sum_{k=1}^{K} |OC^k| = |OB| - |OA| = n - k$ となるため，$U(k)$ に要する記憶容量はほぼ kn となる．

4.2　範囲問合せアルゴリズム

図 2 に maesa の範囲問合せアルゴリズムを示す[18]．このアルゴリズムの中心となる関数は，事例 p，q 間の距離の精密な下限を導く $lb(v_p^{(k)}, v_q^{(k)}, e_p^{(k)}, e_q^{(k)})$ である．この下限値は Theorem 2 に基づく以下の式から得られる．

$$lb(v_p^{(k)}, v_q^{(k)}, e_p^{(k)}, e_q^{(k)}) = \hat{x}_{p,q}^{(k)} - \Delta x_{p,q}^{(k)} \tag{17}$$

ただし，$\hat{x}_{p,q}^{(k)} = {v_p^{(k)}}^T v_q^{(k)}$，$\Delta x_{p,q}^{(k)} = \sqrt{e_p^{(k)}}\sqrt{e_q^{(k)}}$ である．また，$v_p^{(k)}$ と $v_q^{(k)}$ は u_p，u_q から式 (10)，(11) の定義を用いて得られる．

distance computing 段階では，与えられた問合せ事例 q と順序付きリスト OA 内の k ステップのピボット事例間の距離を直接計算し，*EC-transform*(*SG-transform*) を施した後にベクトル $\boldsymbol{u}_q$ に加える．さらに，q からの距離が与えられた範囲 r よりも近ければ，ピボット事例を範囲問合せ結果に加える．次に updating lower bound 段階において，$\boldsymbol{u}_q$ を前処理の k ステップ目で因数分解された $U[k]$ に加える．その後，不完全コレスキー分解を 1 ステップ実行する．$U[k]$ は途中まで因数分解された行列であるため，この 1 ステップ分解は式 (5) を用いて行う．さらに，式 (6) により $e_q^{(k)}$ を計算する．その後，updating query result 段階において，前述の精密な下限値計算と *EC-transform*(*SG-transform*) による EDM 領域への逆変換が行われる．この下限値の情報により，問合せ事例 q から遠い事例を判別，除外し，候補事例集合の中から q に近い事例のみを Q に保存する．この操作はあらかじめ設定した最大ステップ数まで繰り返される．その後，distance computing の段階で，ピボット事例を除く少数の事例に対して直接距離計算が行われ，最後に finalizing query result において，事例 q から距離 r 以内の範囲問合せ事例集合 Q が得られる．

図 2 の範囲問合せアルゴリズム全体の計算時間複雑性は $O(km+kn)$ である．また，ループ内の唯一の直接計算から，問合せ複雑性は $O(km)$ となる．最初の節で言及した範囲問合せアルゴリズム *laesa* および本提案手法の *maesa* の事前構成アルゴリズムの計算時間複雑性，範囲問合せ複雑性，範囲問合せアルゴリズムの計算時間複雑性は，それぞれ $O(kmn)$，$O(km)$，$O(km+kn)$ であり，提案手法と同一である．これは両手法とも問合せ事例とその他の事例間の距離下限の推定原理が異なるのみで，それ以外の approximation や ellimination などのアルゴリズムの構造は同一なためである．また，これらは *M-tree* の事前構成アルゴリズムの計算時間複雑性 $O(m \log n)$〜$O(mn^\alpha)$ や範囲問合せアルゴリズムの計算時間複雑性 $O(m \log n)$〜$O(mn)$ とそれぞれおおむね同等である．一方，範囲問合せ複雑性 $O(km)$ は，*M-tree* の $O(m \log n)$〜$O(mn)$ よりも明らかに勝る．これに対し，*maesa* と *laesa* の記憶容量複雑性は kn で同じである一方で，*M-tree* は n であり，我々の手法には空間複雑性において難点がある．しかし，後述のとおり，*maesa* で必要とする k は，EDM のランクよりも実験的にはるかに小さい．それゆえ，記憶容量複雑性 kn は，現実的には多くの場合，問題にならない．

5. 評価実験結果

我々は，提案するEDM推定の性能を，効率性と精度および *EC-transform* と *SG-transform* の比較の観点から評価する．また，提案する範囲問合せ手法 *maesa* の種々のパラメータに関する性能を，他の代表的な手法である *laesa*，*M-tree* と比較する．

5.1 EDM 推定の性能評価

3 章で述べたように，*EC-transform* ではユークリッド距離と PSD 類似性尺度の推定値や誤差領域を，PSD 領域だけでなく EDM 領域でも厳密に因数分解可能である．これに対して，*SG-transform* ではテイラー展開と OB 内の事例間距離の 2 乗平均値 $\bar{y}$ の導入による近似的因数分解しかできない．ここではまず，EDM 推定が比較的容易な条件，すなわち事例空間内で事例分布を包摂するのに必要な部分空間の次元 k が小さい人工データについて，高精度な EDM 推定を行わせる評価実験を行い，厳密な *EC-transform* と近似の *SG-transform* が，恵まれた条件下で実用的な推定精度や計算時間をもたらすかを検証する．そのため，人工データとして距離空間内で 10 個のクラスタに属する事例とそれ以外のわずかな背景雑音事例から構成されるものを使用した．この場合，10 個のクラスタ中心を含む部分空間によって多くの事例が包摂されるので，EDM 推定は容易である．実験では，事例数，ベクトル次元数，雑音割合のうち，1 つのパラメータのみを変化させ，残りのパラメータはデフォルト値とした．ただし，各パラメータのデフォルト値は事例数 1,000，ベクトル次元数 1,000，雑音割合 3%とした．高精度な EDM 推定を行うため，許容誤差幅閾値 ε_{tol} は十分に小さい $\varepsilon_{tol} = \bar{y} \times z$（$z = 5\%$）とした．ただし，平均距離 $\bar{y}$ は OB 内の 1,000 個の事例ペアをランダムサンプリングして計算した．また，*SG-transform* のパラメータ λ は，予備実験により計算時間と精度のバランスから $\lambda = 1 \times 10^{-5}$ が最適と判断して用いた．

表 1 に，事例数を変化させたときの *EC-transform* と *SG-transform*，距離をすべて直接計算した場合の性能比較を示す．事前構成計算時間は，ピボット事例の選択など推定の準備構成に要する時間である．また，推定計算時間は，推定値の算出に要する時間である．表において，最も良い性能を示した値をイタリック体で表記する．表 1 より，*EC-transform* の推定計算時間は直接計算と比較して，十分に高速であることが分かる．また，推定誤差は許容誤差 ε_{tol} を遥かに下回りきわめて良好である．これに対して，*SG-transform* の計算時間も *EC-transform* と同程度に高速であるが，誤差の標準偏差は巨大である．この値も本来は 5% 以下になるはずであるが，*SG-transform* では式 (16) の近似により，PSD 類似性尺度からユークリッド距離へ逆変換の際，推定値の誤差が許容し難い大きさとなった．以上より，EDM 要素と PSD 要素間の変換関数の選択は，EDM 推定においてきわめて重要であり，よく用いられる式 (16) のようなガウスカーネル関数では誤差幅が因数分解困難なため，良い結果をもたらさないことが分かる．さらに，その他のパラメータを変化させたデータに関する実験結果も同様の傾向を示した．そこで，以降では，*EC-transform* を用いた場合のみを性能評価する．

表 1　事例数に対する *EC*・*SG-transform* の性能評価
Table 1　Performance of *EC*・*SG-transform* against number of objects.

事例数	100	300	1,000	3,000	10,000
事前構成	0.016	*0.078*	0.890	7.53	145
計算時間（sec）	*0.015*	0.093	*0.609*	*4.17*	*24.3*
推定計算	8.6×10^{-4}	*0.010*	*0.172*	*5.86*	464.
時間（sec）	7.5×10^{-4}	0.016	0.188	6.81	*441.*
	0.078	0.812	11.2	117.	1528.
選択された	*13*	20	43	109	462
ピボット事例数	*13*	*19*	*40*	*100*	*168*
誤差の	*0.16*	*0.34*	*0.6*	*0.4*	*0.25*
標準偏差（%）	2×10^3	6×10^3	2×10^4	6×10^4	2×10^5

上段：*EC-transform*/下段：*SG-transfrom*，推定計算時間のみ
上段：*EC-transform*/中段：*SG-transfrom*/下段：直接計算，
誤差の標準偏差は $\bar{y}$ を 100%としている．

表 2　雑音割合に対する *EC-transform* の性能評価
Table 2　Performance of *EC-transform* against noise ratio.

雑音割合（%）	0	3	6	10	100
事前構成	0.218	0.890	1.17	2.11	20.2
計算時間（sec）					
推定計算	*0.006*	*0.017*	*0.020*	*0.042*	*0.283*
時間（sec）	1.12	1.12	1.12	1.12	1.12
選択された	11	43	73	114	944
ピボット事例数					
誤差の標準偏差（%）	0.46	0.60	0.55	0.54	0.57

推定計算時間のみ上段：*EC-transform*/下段：直接計算，
誤差の標準偏差は $\bar{y}$ を 100%としている．

表 2 は，雑音割合を変化させたときの *EC-transform* と直接計算の性能比較を示す．全体に雑音割合が増加すると，事例分布を包摂するのに必要な部分空間の次元 k，すなわちピボット事例数も増加するため，計算時間も増加する．しかし，最右列の雑音割合 100%で事例がまったくランダムに分布するデータの場合でも，直接計算に比べ推定計算時間は短く，

表 3　人工データに関する平均範囲問合せ時間比較
Table 3　Comparison of average range query time for artificial datasets.

許容誤差幅閾値（%）	3	10	30	100	300
平均範囲	*94.7*	*84.7*	*68.7*	*49.1*	*37.1*
問合せ時間	226.	226.	226.	226.	226.
（$\times 10^{-3}$sec）	120.	120.	120.	120.	120.
ベクトル次元数	100	300	1,000	3,000	5,000
平均範囲	*5.86*	*32.5*	*49.1*	*82.7*	*120.*
問合せ時間	21.8	124.	226.	559.	939.
（$\times 10^{-3}$sec）	106.	83.9	120.	237.	430.
雑音割合（%）	0	3	6	10	100
平均範囲	*19.2*	*49.1*	*53.8*	*66.5*	*174.*
問合せ時間	103.	226.	345.	407.	366.
（$\times 10^{-3}$sec）	58.9	120.	144.	163.	419.
問合せ距離（%）	1	3	5	10	30
平均範囲	*29.4*	*36.2*	*49.1*	*49.3*	50.0
問合せ時間	157.	188.	226.	278.	282.
（$\times 10^{-3}$sec）	122.	121.	120.	112.	*17.8*

上段：*maesa*/中段：*laesa*/下段：*M-tree*

事前構成計算時間を含めても大幅な計算時間増加にはならないことが分かる．

5.2　*maesa* の性能評価

ここではまず前節と同様な人工データを使用して評価を行う．ただし，事例数のみ 1×10^4 に固定し，許容誤差幅閾値 ε_{tol}，ベクトル次元数，雑音割合および範囲問合せ距離 r を変化させた．なお，後に示す理由により ε_{tol} のデフォルト値は，事例間距離の 2 乗の平均値 $\bar{y}$ の 100%とした．また，r のデフォルト値は事例間平均距離の 5%とした．

表 3 に，デフォルトデータから各種条件を変化させた場合の *maesa* と既存手法 *laesa*，*M-tree* について，各々10 万回の範囲問合せ試行の 1 回あたり平均範囲問合せ時間の比較を示す．許容誤差幅閾値 ε_{tol} のみを 3%から 300%まで変化させた場合の範囲問合せ時間は，いずれも既存手法に比べ十分に高速である．特に許容誤差条件の緩和による不完全コレスキー分解の早期終了と，距離下限値の減少による非類似事例の elimination 効率低下のトレードオフの下で，ε_{tol} が大きいほど高速になることが分かる．**表 4** に 1%〜300%の ε_{tol} に関して，*maesa* の事例間距離下限の平均値を示す．次元の呪いにより問合せの入力事例と他の大半の事例との距離はほぼ $\bar{y}$ となるため，下限値の推定精度が十分高ければ，その平均は $\bar{y}$ をわずかに下回る程度の値となるはずである．実際に表 4 では，$\varepsilon_{tol} = 100\%$以下において *maesa* の下限値は $\bar{y}$ よりわずかに小さいだけである．これは，不完全コレスキー分解が事例空間中で事例分布を包摂する主要な基底成分から先に分解するため，許容誤差幅閾値 ε_{tol} が事例間 2 乗平均距離程度に大きくても誤差領域の幅が十分小さくなるためである．このように *maesa* では，範囲問合せ距離 r が $\bar{y}$ より小さければ，下限値が r よりも大きいほとんどの事例が効率的に elimination により除去されることが分かる．これに対して，表 4 の脚注に示す *laesa* の下限平均値は *maesa* の約 4 分の 1 であり，r が $\bar{y}$ の 25%以上の場合にはほとんど距離を直接計算するのと変わらなくなってしまう．以上の知見を基に，効率性の観点から *maesa* では ε_{tol} を 100%とした．

表 4　人工データに関する事例間距離下限の平均値比較
Table 4　Comparison of averaged distance lower bounds between objects for artificial datasets.

許容誤差幅閾値（%）	1	10	100	150	300
maesa（%）	98.4	98.0	98.1	63.0	25.5

laesa の下限平均値：25.6%，$\bar{y}$ を 100%としている．

一方，表 3 に示されるように，ベクトル次元数が増加すると，事例間距離下限値判別による直接距離計算の節減効果が次元の呪いにより減少することや個々の事例間距離の計算時間の増加により，いずれの手法とも範囲問合せ時間は増加する．また，雑音割合が増加すると，事例分布を包摂するのに必要なピボット事例数の増加により，いずれの手法でも同じく範囲問合せ時間は増加する．さらに範囲問合せ距離 r を増やすと，*maesa* と *laesa* では問合せ範囲に該当する事例数の増加により範囲問合せ時間も増加する．ただし，*M-tree* では，r があらかじめ階層的に分類された事例クラスタの直径を超えると，クラスタ中心が十分問合せ事例に近い事例は，距離計算なしにすべて問合せ結果に含めることができるのでかえって高速化する．しかし，r が事例間平均距離の数十 % を超えるような応用は少なく，実際的条件では *maesa* が既存手法に比較してきわめて効率的な範囲問合せを実現することが分かる．

さらに本評価実験では，UCI Machine Learning Repository [28] の ionosphere（事例数 351，ベクトル次元数 34），spambase（事例数 4,601，ベクトル次元数 58），musk（事例数 6,598，ベクトル次元数 167），isolet（事例数 6,238，ベクトル次元数 618）による実データ性能評価を行った．**表 5** に 3%から 300%までの許容誤差幅閾値 ε_{tol} に関する *maesa* の平均範囲問合せ時間と *laesa*，*M-tree* のそれらの比較を示す．人工データによる評価結果と同様に，*maesa* は ε_{tol} が 100%の場合に最速の問合せを実現している．またその場合，*laesa*，*M-tree* と比較しても，spambase を除いて十分に高速であることが分かる．spambase は，事例空間の 1 カ所に密集した多くの事例と他に点在するいくつかの事例により構成される

表 5　実データに関する平均範囲問合せ時間比較
Table 5　Comparison of average range query time for read datasets.

許容誤差幅閾値（%）		3	10	30	100	300
iono-sphere	*maesa*	0.20	0.15	0.10	*0.05*	0.09
	laesa: 0.24			*M-tree*: 5.97		
spam-base	*maesa*	2.97	2.97	2.98	2.95	2.95
	laesa: 2.75			*M-tree*: *1.47*		
musk	*maesa*	6.15	3.91	2.17	*0.95*	1.17
	laesa: 5.05			*M-tree*: 13.0		
isolet	*maesa*	43.7	17.2	4.14	*1.10*	5.05
	laesa: 11.2			*M-tree*: 20.7		

時間の単位は（$\times 10^{-3}$sec）.

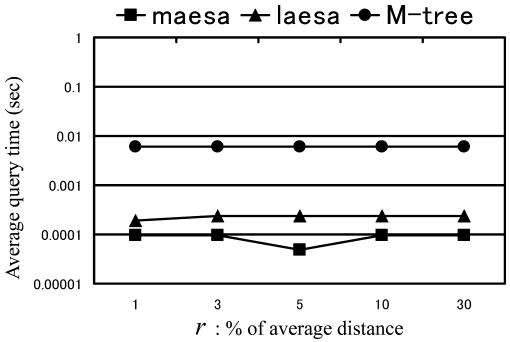

図 3　ionosphere に関する平均範囲問合せ時間比較
Fig. 3　Comparison of average range query time for ionosphere.

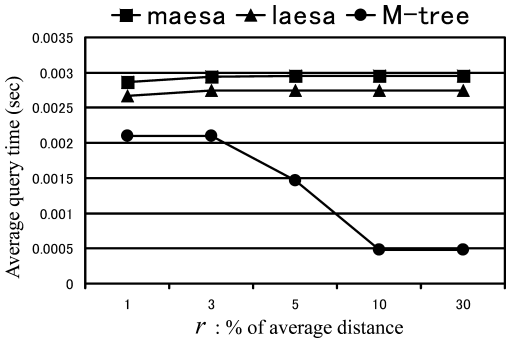

図 4　spambase に関する平均範囲問合せ時間比較
Fig. 4　Comparison of average range query time for spambase.

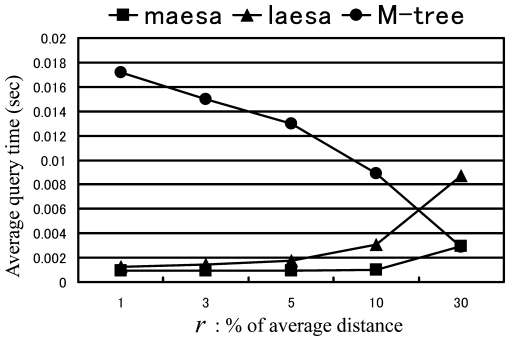

図 5　musk に関する平均範囲問合せ時間比較
Fig. 5　Comparison of average range query time for musk.

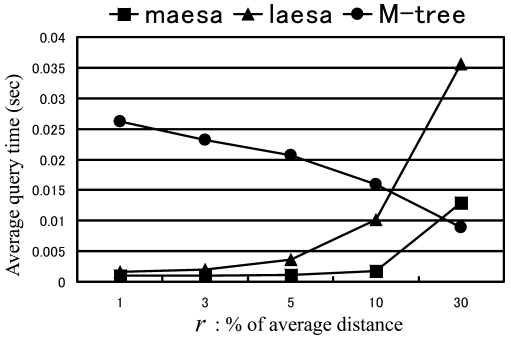

図 6　isolet に関する平均範囲問合せ時間比較
Fig. 6　Comparison of average range query time for isolet.

データである．このため，実験で使用した問合せ事例から問合せ距離閾値内にほぼすべての事例が存在し，*maesa* において直接距離計算を行わなければならない事例が非常に多いため，より多くの計算時間を要している．

図 3，**図 4**，**図 5**，**図 6** に $\varepsilon_{tol} = 100\%$の条件下で，1%から 30%の範囲問合せ距離 r を用いた場合の *maesa* と他既存手法による範囲問合せ時間の比較を示す．図 3 のみ，手法による時間差が比較的大きいため片対数でプロットした．spambase に関しては，先に述べた理由により *maesa* の問合せ時間が *laesa* や *M-tree* に比べ若干遅いが，それ以外では，musk

と isolet のような大規模次元データおよび ionosphere のような小規模次元データのいずれにも，範囲問合せ距離 r がおおむね 30%以下の場合には，本提案手法は既存手法より高速であることが分かる．

6. 結　　論

我々は，近年のデータの大規模化にともなうデータ分析の困難性の増大を軽減するため，ユークリッド距離の高速高精度推定アルゴリズムとそれを応用したユークリッド距離に関する高速な範囲問合せ原理，アルゴリズムを提案した．我々の提案した EDM 推定手法は，精密な誤差幅をともない正確にユークリッド距離を推定する．この研究は，範囲問合せ問題にとどまらず，今後さらに広い理論拡張や応用が期待される．

参 考 文 献

1) Alfakih, A.Y., Khandani, A. and Wolkowicz, H.: Solving euclidean distance matrix completion problems via semidefinite programming, *Computational Optimization and Applications*, Vol.12, No.1-3, pp.13–30 (1999).
2) Artin, E.: *Geometric Algebra*, *Chapter IV*, Interscience (1957).
3) Bach, R. and Jordan, M.I.: Predictive low-rank decomposition for kernel methods, *Proc. ICML*, *The 22nd International Conference on Machine Learning*, pp.9–22 (2005).
4) Baeza-Yates, R., Cunto, W., Manber, U. and Wu, S.: Proximity matching using fixed-queries trees, *Proc. 5th Conference on Combinatorial Pattern Matching* (*CPM'94*), Lecture Notes in Computer Science, 807, pp.198–212, Springer, Berlin Heidelberg New York (1994).
5) Bhatia, R.: *Positive definite matrices*, *Princeton Series in Applied Mathematics*, Princeton University Press, Princeton, New Jersey (2007).
6) Brin, S.: Near neighbor search in large metric spaces, *Proc. VLDB'95*, pp.574–584 (1995).
7) Chavez, E., Marroquin, J. and Baeza-Yates, R.: Spaghettis: An array-based algorithm for similarity qyeries in metric space, *Proc. 6th South American Symposium on String Processing and Information Retrieval* (*SPIRE'99*), IEEE, New York, pp.38–46 (1999).
8) Chavez, E. and Navarro, G.: An effective clustering algorithm to index high dimensional metric spaces, *Proc. 7th South American Symposium on String Processing and Information Retrieval* (*SPIRE'00*), IEEE, NewYork, pp.75–86 (2000).
9) Chavez, E., Marroquin, J. and Navarro, G.: Fixed queries array: A fast and economical data structure for proximity searching, *Multimedia Tools Appl.*, Vol.14, No.2, pp.113–135 (2001).
10) Ciaccia, P., Patella, M. and Zezula, P.: M-tree: An efficient access method for similarity search in metric space, *Proc. 23rd Conference on Very Large Databases* (*VLDB'97*), pp.426–435 (1997).
11) Chavez, E., Navarro, G., Baeza-Yates, R. and Marroquin, J.: Searching in metric spaces, Technical Report TR/DCC-99-3, Dept. of Computer Science, Univ. of Chile (1999).
12) Chavez, E., Navarro, G., Baeza-Yates, R. and Marroquin, J.: Searching in metric space, *ACM Comput. Surv.*, Vol.33, No.3, pp.273–321 (2001).
13) Debnath, A.K., et al.: Structure-activity relationship of mutagenic aromatic and heteroaromatic nitro compounds, Correlation with molecular orbital energies and hydrophobicity, *Journal of Medicinal Chemistry*, Vol.34, pp.786–797 (1991).
14) Dehne, F. and Nolteimer, H.: Voronoi trees and clustering problems, *Information Systems*, Vol.12, No.2, pp.171–175 (1987).
15) Dehne, F.K.H.A. and Noltemeier, H.: Voronoi trees and clustering problems, *Information Systems*, pp.185–194 (1988).
16) Fine, S. and Scheinberg, K.: Efficient SVM training using low-rank kernel representations, *J. Machine Learning Research*, Vol.2, pp.243–264 (2001).
17) Graepel, T.: Kernel matrix completion by semidefinite programming, *Artificial Neural Networks-ICANN 2002*, Dorronsoro, J.R. (Ed.), pp.687-693, Springer Verlag (2002).
18) Kido, K., Kuwajima, H. and Washio, T.: A Range Query Approach for High Dimensional Euclidean Space Based on EDM Estimation, *Proc. 8th SIAM International Conference on Data Mining* (*SDM08*), pp.387–398 (2008).
19) Akaho, S., Tsuda, K. and Asai, K.: The em algorithm for kernel matrix completion with auxiliary data, *J. Machine Learning Research*, Vol.4, pp.67–81 (2003).
20) Kuwajima, H. and Washio, T.: Fast PSD Matrix Estimation by Column Reductions, *ISM Report on Research and Education*, No.25, *The International Workshop on Data-Mining and Statistical Science* (*DMSS2007*), pp.179–189 (2007).
21) Kuwajima, H. and Washio, T.: Fast Approach to Estimate Large PSD Matrices from Partial Data, *Workingnotes of 7th IEEE International Conference on Data Mining – Workshop on High Performance Computing*, IEEE DOI 10.1109/ICDMW.2007.24, pp.337–342 (2007).
22) Laurent, M.: Cuts, matrix completions and graph rigidity, *Mathematical Programming*, Vol.79, pp.255–283 (1997).
23) Laurent, M.: A Connection Between Positive Semidefinite and Euclidean Distance Matrix Completion Problems, *Linear Algebra and its Applications*, Vol.273, pp.9–22

(1998).
24) Laurent, M.: Matrix Complition Problems, *The Encyclopedia of Optimization*, *volume III*, Floudas, C. and Pardalos, P. (Eds.), pp.221–229, Kluwer (2001).
25) Michael, S.T.D., Berry, W. and O'Brien, G.W.: Using linear algebra for intelligent information retrieval, Technical Report: UT-CS-94-270, University of Tennessee. Knoxville, TN, USA (1994).
26) Mico, L., Oncina, J. and Vidal, E.: A new version of the nearest-neighbor approximating and eliminating search (aesa) with linear preprocessing-time and memory requirements, *Pattern Recognition Lett.*, Vol.15, pp.9–17 (1994).
27) Navarro, G.: Searching in metric space by spatial approximation, *The VLDB Journal*, Vol.11, pp.28–46 (2002).
28) Newman, C.B.D.J., Hettich, S. and Merz, C.: Uci repository of machine learning databases, University of California, Irvine, Dept. of Information and Computer Science (1998). http://mlearn.ics.uci.edu/MLRepository.html
29) Sebastiani, F.: Machine learning in automated text categorization, *ACM Computing Surveys*, Vol.34, No.1, pp.1–47 (2002).
30) Tsuda, K., Akaho, S. and Asai, K.: The em Algorithm for Kernel Matrix Completion with Auxiliary Data, *J. Machine Learning Research*, Vol.4, pp.67–81 (2003).
31) Uhlmann, J.: Satisfying general proximity/similarity queries with metric trees, *IPL40*, pp.175–179 (1991).
32) Williams, C.K.I. and Seeger, M.: The effect of the input density distribution on kernel-based classifiers, *Proc. ICML: The 17th International Conference on Machine Learning*, pp.1159–1166 (2000).
33) Yianilos P.N.: Data structures and algorithms for nearest neighbor search in general metric spaces, *Proc. 4th ACM-SIAM Symposium on Discrete Algorithms* (*SODA'93*), pp.311–321 (1993).

(平成 20 年 6 月 27 日受付)
(平成 21 年 2 月 3 日採録)

城戸健太郎（正会員）

1981 年生．2006 年大阪大学工学部電子情報工学科卒業．2008 年大阪大学大学院電気電子情報工学専攻修士課程修了．2008 年兼松株式会社入社．現在に至る．人工知能，データマイニング等の研究を経て，現在は商社業務に従事．人工知能学会会員．

桑島　　洋（正会員）

1983 年生．2006 年東京農工大学工学部情報コミュニケーション工学科卒業．2008 年大阪大学大学院工学研究科電気電子情報工学専攻情報通信工学部門博士前期課程修了．2008 年マイクロソフトディベロップメント（株）入社．現在に至る．人工知能，データマイニング等の研究を経て，現在はソフトウェア開発に従事．人工知能学会会員．

鷲尾　　隆（正会員）

1960 年生．1983 年東北大学工学部原子核工学科卒業．1988 年東北大学大学院原子核工学専攻博士課程修了．工学博士．1988 年から 1990 年にかけてマセチューセッツ工科大学原子炉研究所客員研究員．1990 年（株）三菱総合研究所入社．1996 年退社．大阪大学産業科学研究所助教授（知能システム科学研究部門）．2006 年大阪大学産業科学研究所教授（知能システム科学研究部門）．現在に至る．原子力システムの異常診断手法に関する研究，定性推論に関する研究を経て，現在は人工知能の基礎研究，科学的知識発見，データマイニング等の研究に従事．人工知能学会，計測自動制御学会，日本知能情報ファジイ学会，AAAI，IEEE Computer Society 各会員．