Vol.2013-DBS-157 No.32
Vol.2013-IFAT-111 No.32
2013/7/23

ガウス分布の類似問合せに関する考察

董 テイテイ[1,a)]　肖 川[1,b)]　石川 佳治[1,2,c)]

概要：ガウス分布の類似問合せの処理手法についてアイデアを述べる．類似度としてはカルバック・ライブラー情報量を想定する．問合せ処理が，スカイライン問合せおよびランク集約の考え方を用いて実現できることを示す．

キーワード：ガウス分布，類似問合せ，カルバック・ライブラー情報量，スカイライン問合せ，ランク集約

Similarity Queries on Gaussian Distributions

Tingting Dong[1,a)]　Chuan Xiao[1,b)]　Yoshiharu Ishikawa[1,2,c)]

Abstract: We describe ideas for similarity query processing on Gaussian distributions. We assume the use of Kullback-Leibler divergence as the similarity measure. We show that queries can be processed using the notions of skyline queries and rank aggregation.

Keywords: Gaussian distributions, similarity queries, Kullback-Leibler divergence, skyline queries, rank aggregation

1. はじめに

ガウス分布（Gaussian distribution）は代表的な確率分布の一つである [2]．本研究では，ガウス分布の類似問合せについて議論する．ここでの想定は，データベース中に多数のガウス分布オブジェクトが蓄積されているというものである．ガウス分布は確率分布であるため，類似問合せでは確率分布の類似度を用いることが考えられる．

代表的な確率分布の類似尺度として，**カルバック・ライブラー情報量**（Kullback-Leibler divergence; KL 情報量）がある [7]．本研究では，この KL 情報量の使用を想定する．KL 情報量は距離の公理を満たさない *nonmetric* な尺度である [13]．そのため，M-木のような距離索引 [5], [14] を利用することもできない．

[1] 名古屋大学情報科学研究科
Graduate School of Information Science, Nagoya University
[2] 国立情報学研究所
National Institute of Informatics
[a)] dongtt@db.ss.is.nagoya-u.ac.jp
[b)] chuanx@nagoya-u.jp
[c)] ishikawa@is.nagoya-u.ac.jp

そこで本稿では，ガウス分布の性質と KL 情報量の性質を効果的に用いる．top-k 問合せを対象とし，**スカイライン問合せ**（skyline query）[4] および**ランク集約**（rank aggregation）[8] の考え方を用いた処理手法のアイデアを示す．

2. 問題の定義

2.1 ガウス分布の類似問合せ

1 次元の**ガウス分布**（Gaussian distribution）は，分散を σ^2 とし，平均を μ としたとき，

$$p(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{(x-\mu)^2}{2\sigma^2}\right] \quad (1)$$

と定義される．d 次元ガウス分布は，分散共分散行列を $\boldsymbol{\Sigma}$ とし，平均ベクトルを $\boldsymbol{\mu}$ としたとき，

$$p(\boldsymbol{x}) = \frac{1}{(2\pi)^{d/2}|\boldsymbol{\Sigma}|^{1/2}} \exp\left[-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^t\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right] \quad (2)$$

と定義される [2].

本研究では，多数のガウス分布オブジェクトがデータベースに格納されていると考える．問合せとしてガウス分

布が与えられることを想定し，類似した順に上位 k 件のガウス分布を求める **top-k問合せ**を考える．

2.2 情報量に基づく類似尺度

連続的な確率密度関数に対する**カルバック・ライブラー情報量**（Kullback-Leibler divergence, KL 情報量）あるいは**双対エントロピー**（relative entropy）[7] は，$f(x), g(x)$ を任意の確率密度関数としたとき，

$$\mathcal{D}_{\mathrm{KL}}(f\|g) = \int_{-\infty}^{\infty} f(x) \ln \frac{f(x)}{g(x)} dx \qquad (3)$$

で与えられる．一般に $\mathcal{D}_{\mathrm{KL}}(f\|g) \neq \mathcal{D}_{\mathrm{KL}}(g\|f)$ であり，$\mathcal{D}_{\mathrm{KL}}(f\|g) \geq 0$ である．$\mathcal{D}_{\mathrm{KL}}(f\|g) = 0$ となるのは $f(x) = g(x)$ のときである．情報理論の立場からは，KL 情報量 $\mathcal{D}(f\|g)$ は，真の分布が f のときに分布が g であると仮定した場合の非効率さの測度であると説明される [7]．KL 情報量は三角不等式も満たさず，*nonmetric* な尺度である [13]．

なお，非対称である KL 情報量を類似問合せに用いるには，q, p をそれぞれ問合せ分布，データベース中の分布としたとき，$\mathcal{D}_{\mathrm{KL}}(p\|q)$ と $\mathcal{D}_{\mathrm{KL}}(q\|p)$ のどちらを使用するかが問題となる．ここでは双方とも考慮することにして，

- **タイプ 1 の問合せ**：$\mathcal{D}_{\mathrm{KL}}(q\|p)$ を使用
- **タイプ 2 の問合せ**：$\mathcal{D}_{\mathrm{KL}}(p\|q)$ を使用

という 2 つのタイプを考える．

3. 問合せの分析：1 次元の場合

ここでは，1 次元のガウス分布について，KL 情報量に関する分析を行う．以下が成り立つ．

命題 1 $f(x) = \mathcal{N}(\mu_f, \sigma_f), g(x) = \mathcal{N}(\mu_g, \sigma_g)$ を 2 つの 1 次元ガウス分布とする．$\mathcal{D}_{\mathrm{KL}}(f\|g)$ は

$$\mathcal{D}_{\mathrm{KL}}(f\|g) = \frac{1}{2}\left[\frac{(\mu_f - \mu_g)^2 + \sigma_f^2}{\sigma_g^2} - \ln\frac{\sigma_f^2}{\sigma_g^2} - 1\right] \qquad (4)$$

で与えられる． ■

$\mathcal{D}_{\mathrm{KL}}(f\|g)$ は閉じた形式（closed form）で与えられ，数値積分を用いて計算する必要はないことに注意する．

3.1 タイプ 1 の問合せの分析

タイプ 1 の問合せ $\mathcal{D}_{\mathrm{KL}}(q\|p)$ について考える．問合せが与えられた時点で定数になる項を除き整理すると，以下の**ランク付け関数**（ranking function）が得られる．

定義 1 KL 情報量を用いたタイプ 1 の問合せに対するランク付け関数を以下のように定義する．

$$\mathcal{R}^q_{\mathrm{KL1}}(p) = \frac{(\mu_q - \mu_p)^2 + \sigma_q^2}{\sigma_p^2} + 2\ln\sigma_p \qquad (5)$$

μ_p, σ_p に対する $\mathcal{D}_{\mathrm{KL}}(q\|p), \mathcal{R}^q_{\mathrm{KL1}}(p)$ の増減は一致する． ■

$\mathcal{R}^q_{\mathrm{KL1}}(p)$ は $|\mu_q - \mu_p|$ について単調増加する関数である．σ_p については，以下のような性質を導くことができる．

性質 1 $\mathcal{R}^q_{\mathrm{KL1}}(p)$ は，$\sigma_p^2 \leq \sigma_q^2 + (\mu_q - \mu_p)^2$ において単調減少し，$\sigma_p^2 > \sigma_q^2 + (\mu_q - \mu_p)^2$ において単調増加する． ■

この性質を図示したものが**図 1** である．見やすさのため，横軸は $(\mu_q - \mu_p)^2$，縦軸は σ_p^2 としている．矢印は KL 情報量が減少する方向（値が良くなる方向）を指している．図より，ランク付け関数は区分的に単調性があることがわかる．

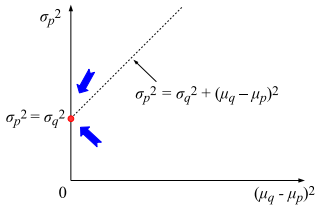

図 1 $\mathcal{R}^q_{\mathrm{KL1}}(p)$ の性質
Fig. 1 Property of $\mathcal{R}^q_{\mathrm{KL1}}(p)$

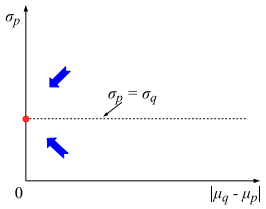

図 2 $\mathcal{R}^q_{\mathrm{KL2}}(p)$ の性質
Fig. 2 Property of $\mathcal{R}^q_{\mathrm{KL2}}(p)$

3.1.1 タイプ 2 の問合せの分析

ランク付け関数は次のように定義できる．

定義 2 ランク付け関数を以下のように定義する．

$$\mathcal{R}^q_{\mathrm{KL2}}(p) = (\mu_p - \mu_q)^2 + \sigma_p^2 - 2\sigma_q^2 \ln\sigma_p \qquad (6)$$

μ_p, σ_p に対する $\mathcal{D}_{\mathrm{KL}}(p\|q), \mathcal{R}^q_{\mathrm{KL2}}(p)$ の増減は一致する． ■

σ_p について次のような性質が成り立つ．

性質 2 $\mathcal{R}^q_{\mathrm{KL2}}(p)$ は，$\sigma_p \leq \sigma_q$ で単調減少し，$\sigma_p > \sigma_q$ で単調増加する． ■

タイプ 1 と異なり，最小値をとる値は $|\mu_q - \mu_p|$ には依存しない．**図 2** に性質を図示する．

3.2 スカイライン問合せとしての解釈

KL 情報量のタイプ 1 の問合せを考える．**図 3** のようにガウス分布オブジェクトが分布しているとする．c と d を比べると，c の方が μ_q からの平均の位置が遠く，分散が小さい．斜線の下側の領域では，分散が小さいほど不利なため，c の方が KL 情報量の値が悪い（より大きい）と判断できる．同様に f, g も d に支配される．斜線の上側の領域でも，支配の方向は違うが同じように考えることができ，m は o, r, s, t, u を支配する．

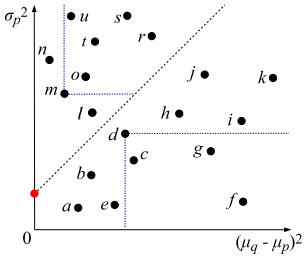

図 3 オブジェクトの分布例
Fig. 3 Example of objects

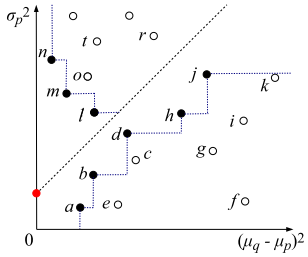

図 4 スカイライン問合せの結果
Fig. 4 Skyline query result

なお，それぞれのガウス分布オブジェクトがどちらの領

域に属するかは，簡単な条件で判定できる．特に KL 情報量のタイプ 2 の場合は簡単になる．

図 3 に対する**スカイライン問合せ**（skyline query）[4] の結果を**図 4** に示す．$\{a, b, d, h, j\}$ と $\{l, m, n\}$ が各領域のスカイラインである．最も KL 情報量の値が良いオブジェクトは，スカイラインを構成するオブジェクトのいずれかである点に注意する．つまり，これら 8 個のオブジェクトについて KL 情報量を計算し，最も値の良いものが最適解（top-1）となる．

top-2 のオブジェクトを見つける場合には，まず top-1 のオブジェクトをスカイラインから削除し，スカイラインを再構成する．スカイラインを構成するオブジェクトの中から最も KL 情報量の値が良いものを選べばよい．

3.3 問合せ処理アルゴリズム

KL 情報量のタイプ 1 の問合せを例にとり，問合せ処理の大まかなアイデアを説明する．アルゴリズムの概略を**図 5** に示す．

```
1: function KLD_Query(q)                          ▷ for 1-D & type 1 case
2:     S ← ∅;                                     ▷ Init the result set
3:     u_graph ← ∅; l_graph ← ∅;                  ▷ Dominance graphs
4:     initNNQuery(μ_q, 0);                       ▷ Initialize NN query
5:     loop
6:         (μ_p, σ_p) ← nextNN();
7:         if σ_p^2 ≥ σ_q^2 + (μ_q − μ_p)^2 then
8:             u_graph.add(μ_p, σ_p);             ▷ Add to the upper graph
9:         else
10:            l_graph.add(μ_p, σ_p);             ▷ Add to the lower graph
11:        end if
12:        if stop condition (*) is satisfied then
13:            Compute KL divergence for each skyline object;
14:            Add the best object to S;
15:            if |S| = k then
16:                return S;
17:            end if
18:            Remove the best object from u_graph or l_graph;
19:        end if
20:    end loop
21: end function
```

図 5 問合せ処理アルゴリズム（1-D，タイプ 1）
Fig. 5 Query processing algorithm (1-D, type 1)

事前に各ガウス分布オブジェクト (μ_p, σ_p) を 2 次元の空間索引に登録しておく．関数 initNNQuery() と nextNN() により空間索引を用いた最近傍探索を行う．4 行目で設定しているように，点 $(\mu_q, 0)$ から近い順に検索する．7 行目で，索引から (μ_p, σ_p) が得られたとき，$\sigma_p^2 \geq \sigma_q^2 + (\mu_q - \mu_p)^2$ であれば上部領域，そうでなければ下部領域に属すると判定する．上部，下部それぞれの領域について，支配関係に基づくグラフをインクリメンタルに構築していく（8 行目および 10 行目．詳細は省略）．12 行目で，停止条件が成立しているかどうかを判断している．ここでの停止条件は，「まだ探索していないすべてのオブジェクトが，支配関係グラフにより得られるスカイラインに支配されている」ということである．このチェックは空間的な条件を用いて容易に行える[*1]．停止条件が成り立つと，スカイラインを構成する各オブジェクトについて KL 情報量を求める（13 行目）．計算済の値があればそれを用いる．KL 情報量が最大のオブジェクトを S に追加し，もし S の要素数が k ならば処理を終了する．そうでなければ処理を継続するが，S に追加したオブジェクトを支配グラフから削除しておく（18 行目）．

実装レベルのアルゴリズムでは，さらに検討すべき問題がある．一つには，ここでのスカイライン問合せは $|\mu_q - \mu_p|$ と σ_p の 2 次元空間を対象としているが，探索空間上でのそれぞれの分布パターンが大きく違っている可能性がある．第一に，最近傍探索を行う際に，$\mathcal{D}(\mu_p, \sigma_p) = \sqrt{(\mu_q - \mu_p)^2 + w\sigma_p^2}$ といった重み付きユークリッド距離を用いることが考えられる．正の定数 w をどのように定めるかがポイントとなる．第二に，$|\mu_q - \mu_p|$ は動的に決まるのに対し，σ_p は静的に決まるという違いもあり，その性質をどう活用するかという課題もある．さらに，問題の性質を考慮して索引構造を工夫することも考えられる．

4. 多次元の場合：次元独立の場合

次に，ガウス分布の各次元が独立である場合を考える．このとき，ガウス分布の等距離面は軸並行の楕円体となる．

4.1 問合せの分析

$f(x) = \mathcal{N}(\boldsymbol{\mu}_f, \boldsymbol{\Sigma}_f), g(x) = \mathcal{N}(\boldsymbol{\mu}_g, \boldsymbol{\Sigma}_g)$ を 2 つの d 次元ガウス分布とする．独立性の仮定から，$\boldsymbol{\Sigma}_f, \boldsymbol{\Sigma}_g$ は対角行列となる．対角要素をそれぞれ $\sigma_{f,i}, \sigma_{g,i}$ で，また，平均ベクトルの要素をそれぞれ $\mu_{f,i}, \mu_{g,i}$ で表す $(i = 1, \ldots, d)$.

以下では，KL 情報量のタイプ 1 の問合せを例にとり分析を行う．

命題 2 $\mathcal{D}_{\mathrm{KL}}(f\|g)$ は

$$\mathcal{D}_{\mathrm{KL}}(f\|g) = \frac{1}{2}\left[\sum_{i=1}^{d}\left(\frac{(\mu_{g,i} - \mu_{f,i})^2}{\sigma_{g,i}^2} + \frac{\sigma_{f,i}^2}{\sigma_{g,i}^2} - \ln\frac{\sigma_{f,i}^2}{\sigma_{g,i}^2}\right) - d\right] \quad (7)$$

で与えられる． ■

問合せガウス分布を $q = (\boldsymbol{\mu}_q, \boldsymbol{\Sigma}_q)$ とし，データベース中のあるガウス分布を $p = (\mu_p, \boldsymbol{\Sigma}_p)$ としたときのタイプ 1 の KL 情報量 $\mathcal{D}_{\mathrm{KL}}(q\|p)$ について考える．

定義 3 i 番目の次元についてのランク付け関数は

[*1] ただし，そのためには nextNN() をこのタイミングで呼び出す必要がある．図 5 の表現では，分かりやすさの方を優先した．

$$\mathcal{R}^q_{\mathrm{KL1}}(p,i) = \frac{(\mu_{q,i}-\mu_{p,i})^2+\sigma^2_{q,i}}{\sigma^2_{p,i}} + 2\ln\sigma_{p,i} \qquad (8)$$

となる．■

この式は，i に対する添え字以外は 1 次元の場合と同じであり，増減について同様の性質が成り立つ．

4.2 問合せ処理アルゴリズム

上記の分析により，ある次元について最良のスコアを与えるオブジェクトは，スカイライン問合せにより特定できることになる．しかし，式 (7) を見ると，求める KL 情報量を得るには，各次元のスコアの和をとる必要がある．ある次元で類似したオブジェクトが別の次元で類似しているとは限らないため，問合せ処理において工夫が必要である．

そこで，いわゆる**ランク集約**（rank aggregation）[8] の考え方を用いる．これは，サーベイ論文 [10] では，**top-k 選択問合せ**（top-k selection query）と呼ばれている．ここでは，代表的なアルゴリズムである *threshold algorithm* (TA) [8] を一部修正して用いる．

各次元 i ごとに，$\mathcal{R}^q_{\mathrm{KL1}}(p,i)$ の小さい順にオブジェクト id p と対応する i 次元の平均値 $\mu_{p,i}$ と標準偏差 $\mu_{p,i}$ を返す，**ソートされたアクセス**（sorted access）機能を提供するサブシステム S_i が存在することを想定する．サブシステムでは，3.3 節で述べた問合せ処理アルゴリズムを実装する．一方，オブジェクト ID を与えると，対応する平均ベクトル $\boldsymbol{\mu}_p$ と分散共分散行列 $\boldsymbol{\Sigma}_p$ の値を返す**ランダムアクセス**（random access）機能も提供されるとする．

問合せ処理アルゴリズムを**図 6** に示す．基本的には TA アルゴリズムに基づくが，本研究に応じた修正を行っている．10 行目では，$|Q|<k$ の場合は単なる追加，$|Q|=k$ の場合は追加と同時に Q 中の最下位オブジェクトの削除が行われる．12 行目に出てくる仮想オブジェクト t について説明する．11 行目が終わった時点で，S_i $(i=1,\ldots,d)$ において最後にアクセスした（これまで見たうちで最も劣っている）オブジェクトの i 番目の次元の平均値，標準偏差を $\bar{\mu}_i, \bar{\sigma}_i$ とする．仮想オブジェクト t は，各次元 i $(i=1,\ldots,d)$ の平均値，標準偏差が $\bar{\mu}_i, \bar{\sigma}_i$ であるようなガウス分布である．ここで計算される閾値 τ は，まだ見ていないオブジェクトがとりうる最良のスコアである．Q に含まれるオブジェクトが k 個に達し，各オブジェクトのスコアが τ より良いならば処理を終了する（13 行目）．

TA アルゴリズムによる問合せ処理は**インスタンス最適性**（instance optimality）と呼ばれる効率性が保証されており，理論上は効率的なアルゴリズムである．問合せ処理の具体的な実現方式については今後の課題としたい．特に，問合せガウス分布の標準偏差 $\sigma_{q,i}$ $(i=1,\ldots,d)$ の値が次元による大きく異なっている場合，そのことをアクセスの効率化に活用できる可能性がある．

```
 1: function KLD_Query(q)
 2:     ▷ d-dim case (type 1): Each dimension is independent
 3:   Q ← ∅                                  ▷ Priority queue with size k
 4:   loop                  ▷ Sorted access to S_1, ..., S_d in parallel
 5:     Let p is obtained by accessing S_i;
 6:     if p is accessed for the first time then
 7:       Do a random access to p;
 8:       Compute s ← D_KL(q‖p);
 9:       if the score s is within top-k then
10:         Q.add(⟨p, s⟩);
11:       end if
12:       τ ← D_KL(q‖t);                    ▷ t is a virtual object
13:       if (|Q| = k) ∧ (∀⟨p, s⟩ ∈ Q, s ≤ τ) then
14:         return Q;
15:       end if
16:     end if
17:   end loop
18: end function
```

図 6 問合せ処理アルゴリズム（d-D，タイプ 1，次元独立）
Fig. 6 Query procesing algorithm (d-D, type 1, each dimension is independent)

5. 多次元の場合：一般の場合

一般の場合，すなわち分散共分散行列が対角行列でない場合について考える．

5.1 定義と分析の準備

KL 情報量は以下のように与えられる．

命題 3 $\mathcal{D}_{\mathrm{KL}}(f\|g)$ は

$$\mathcal{D}_{\mathrm{KL}}(f\|g) = \frac{1}{2}\left[\ln\frac{\det(\boldsymbol{\Sigma}_g)}{\det(\boldsymbol{\Sigma}_f)} + \mathrm{tr}(\boldsymbol{\Sigma}_g^{-1}\boldsymbol{\Sigma}_f) + (\boldsymbol{\mu}_g-\boldsymbol{\mu}_f)^t\boldsymbol{\Sigma}_g^{-1}(\boldsymbol{\mu}_g-\boldsymbol{\mu}_f) - d\right] \qquad (9)$$

で与えられる．ただし，行列 $\mathbf{M}$ に対し，$\det(\mathbf{M})$ は $\mathbf{M}$ の行列式であり，$\mathrm{tr}(\mathbf{M})$ は $\mathbf{M}$ のトレースである．■

この場合，これまでの分析手法を適用することは困難である．分散共分散行列の d^2 個の要素を考える必要があり（実際には分散共分散行列は対称であるため約半分となる），次元間の相関も存在する．そこで，近似によるフィルタリング処理を考える．

与えられたガウス分布 $\mathcal{N}(\boldsymbol{\mu},\boldsymbol{\Sigma})$ に対し，$\boldsymbol{\Sigma}^{-1}$ を，

$$\boldsymbol{\Sigma}^{-1} = \sum_{i=1}^{d}\lambda_i\boldsymbol{v}_i\boldsymbol{v}_i^t \qquad (10)$$

とスペクトル分解する．$\lambda_i, \boldsymbol{v}_i$ はそれぞれ，i 番目の固有値と固有ベクトルである．なお，分散共分散行列 $\boldsymbol{\Sigma}^{-1}$ については固有値は正の値をとる．また，

$$\lambda_{\min} = \min\{\lambda_i\} \qquad (11)$$

$$\lambda_{\max} = \max\{\lambda_i\} \qquad (12)$$

とおく．

5.2 問題の分析（タイプ 1）

まず，タイプ 1 の問合せ $\mathcal{D}_{\mathrm{KL}}(q\|p)$ について考える．

命題 4 ランク付け関数 $\mathcal{R}^q_{\mathrm{KL1}}(p)$ は

$$\mathcal{R}^q_{\mathrm{KL1}}(p) = \ln(\det(\boldsymbol{\Sigma}_p)) + \mathrm{tr}(\boldsymbol{\Sigma}_p^{-1}\boldsymbol{\Sigma}_q) + (\boldsymbol{\mu}_p - \boldsymbol{\mu}_q)^t\boldsymbol{\Sigma}_p^{-1}(\boldsymbol{\mu}_p - \boldsymbol{\mu}_q) \quad (13)$$

で与えられる． ■

5.3 近似関数を用いた問合せ処理（タイプ 1）

近似的なランク付け関数を以下のように定義する．

命題 5 近似的なランク付け関数 $\underline{\mathcal{R}}^q_{\mathrm{KL1}}(p)$ を，

$$\underline{\mathcal{R}}^q_{\mathrm{KL1}}(p) = \ln(\det(\boldsymbol{\Sigma}_p)) + \lambda^p_{\min}\left[\mathrm{tr}(\boldsymbol{\Sigma}_q) + \|\boldsymbol{\mu}_p - \boldsymbol{\mu}_q\|^2\right] \quad (14)$$

で与える．ただし，$\lambda^p_{\min}$ は $\boldsymbol{\Sigma}_p^{-1}$ に対する $\lambda_{\min}$ であり，$\|\boldsymbol{\mu}_p - \boldsymbol{\mu}_q\|$ は $\boldsymbol{\mu}_p, \boldsymbol{\mu}_q$ のユークリッド距離である． ■

$\underline{\mathcal{R}}^q_{\mathrm{KL1}}(p)$ について以下の性質が成り立つ．

定理 1 常に $\underline{\mathcal{R}}^q_{\mathrm{KL1}}(p) \le \mathcal{R}^q_{\mathrm{KL1}}(p)$ が成立する． ■

証明： $\mathbf{A}, \mathbf{B}$ が $d \times d$ の半正定値（positive semidefinite）行列であるとき，$\mathrm{tr}(\mathbf{AB}) \ge \lambda_{\min}(\mathbf{A})\mathrm{tr}(\mathbf{B})$ が成り立つ [9]．ただし，$\lambda_{\min}(\mathbf{A})$ は $\mathbf{A}$ の最小固有値である．分散共分散行列およびその逆行列は半正定値行列であるので，$\mathbf{A} = \boldsymbol{\Sigma}_p^{-1}, \mathbf{B} = \boldsymbol{\Sigma}_q$ と置くと $\mathrm{tr}(\boldsymbol{\Sigma}_p^{-1}\boldsymbol{\Sigma}_q) \ge \lambda^p_{\min}\mathrm{tr}(\boldsymbol{\Sigma}_p)$ となる．$(\boldsymbol{\mu}_p - \boldsymbol{\mu}_q)^t\boldsymbol{\Sigma}_p^{-1}(\boldsymbol{\mu}_p - \boldsymbol{\mu}_q) \ge \lambda^p_{\min}\|\boldsymbol{\mu}_p - \boldsymbol{\mu}_q\|^2$ が成り立つことは [11] による． □

つまり，$\underline{\mathcal{R}}^q_{\mathrm{KL1}}(p)$ は実際の $\mathcal{R}^q_{\mathrm{KL1}}(p)$ の値よりもより良い値を返すことがあるが，その逆はない．この性質の利用について述べる．データベースの各ガウス分布 $\mathcal{N}(\boldsymbol{\mu}_p, \boldsymbol{\Sigma}_p)$ について，$\boldsymbol{\mu}_p$ の情報だけでなく，行列式 $\det(\boldsymbol{\Sigma}_p)$ および $\boldsymbol{\Sigma}_p^{-1}$ の最小固有値 $\lambda^p_{\min}$ の情報を事前計算する．$\underline{\mathcal{R}}^q_{\mathrm{KL1}}(p)$ は $\det(\boldsymbol{\Sigma}_p), \lambda^p_{\min}, \|\boldsymbol{\mu}_p - \boldsymbol{\mu}_q\|$ のいずれに対しても単調増加関数であるので，4 章の考え方を用いると，これらの 3 つの次元に対するスカイライン問合せにより，$\underline{\mathcal{R}}^q_{\mathrm{KL1}}(p)$ の小さい順にオブジェクトを検索できる．

問合せ処理アルゴリズムを**図 7** に示す．5 行目の関数 nextAprNN(q) は，$\underline{\mathcal{R}}^q_{\mathrm{KL1}}(p)$ が小さい順にオブジェクト ID と近似スコアを返す関数である．その実装は，4 章のアプローチで実現できる．得られたオブジェクト p に対し，真のランク付け関数の値 $s = \mathcal{R}^q_{\mathrm{KL1}}(p)$ を求めて優先度付きキュー Q に追加していく．ただし，Q のサイズは k であり，余分なオブジェクトはキューから削除される．6 行目では終了条件を判定している．Q に k 個の要素が含まれており，Q 中の最大（つまり k 番目）のランク付け関数のスコアが，nextAprNN(q) からいま得られたオブジェクトの近似スコア以下であれば，それ以降処理を続ける必要はないため，アルゴリズムを終了する．

5.4 タイプ 2 の場合の処理

$\mathcal{D}_{KL}(p\|q)$ の場合も同様の考え方で処理できる．ランク付け関数を以下のように定義する．

```
1: function KLD Query(q)        ▷ d-dim general case (type 1)
2:     Q ← ∅;                   ▷ Priority queue with size k;
3:     max_s ← 0;
4:     loop
5:         ⟨p, s⟩ ← nextAprNN(q)   ▷ NN query using R̲^q_KL1(p)
6:         if (|R| = k) ∧ (s ≥ max_s) then
7:             return Q;            ▷ No more good candidates
8:         end if
9:         s ← R^q_KL1(p);          ▷ Compute the true score
10:        R.push(⟨p, s⟩);          ▷ Add to the queue
11:        max_s ← max{s, max_s}
12:    end loop
13: end function
```

図 7 一般の場合の問合せ処理アルゴリズム
Fig. 7 Query processing algorithm for general case

命題 6 ランク付け関数 $\mathcal{R}^q_{\mathrm{KL2}}(p)$ を，

$$\mathcal{R}^q_{\mathrm{KL2}}(p) = -\ln(\det(\boldsymbol{\Sigma}_p)) + \mathrm{tr}(\boldsymbol{\Sigma}_q^{-1}\boldsymbol{\Sigma}_p) + (\boldsymbol{\mu}_q - \boldsymbol{\mu}_p)\boldsymbol{\Sigma}_q^{-1}(\boldsymbol{\mu}_q - \boldsymbol{\mu}_p) \quad (15)$$

で与える． ■

近似的なランク付け関数を以下のように定義する．

命題 7 近似的なランク付け関数 $\underline{\mathcal{R}}^q_{\mathrm{KL2}}(p)$ を，

$$\underline{\mathcal{R}}^q_{\mathrm{KL2}}(p) = d\ln\lambda^p_{\min} + \frac{1}{\lambda^p_{\max}}\mathrm{tr}(\boldsymbol{\Sigma}_q^{-1}) + (\boldsymbol{\mu}_q - \boldsymbol{\mu}_p)\boldsymbol{\Sigma}_q^{-1}(\boldsymbol{\mu}_q - \boldsymbol{\mu}_p) \quad (16)$$

で与える．ただし，$\lambda^p_{\min}, \lambda^p_{\max}$ は $\boldsymbol{\Sigma}_p^{-1}$ に対する $\lambda_{\min}, \lambda_{\max}$ である． ■

$\underline{\mathcal{R}}^q_{\mathrm{KL2}}(p)$ について以下の性質が成り立つ．

定理 2 常に $\underline{\mathcal{R}}^q_{\mathrm{KL2}}(p) \le \mathcal{R}^q_{\mathrm{KL2}}(p)$ が成立する． ■

証明： $\boldsymbol{\Sigma}_p^{-1}$ の固有値を $\lambda^p_1, \ldots, \lambda^p_d$ とすると，$\boldsymbol{\Sigma}_p$ の固有値はそれぞれの逆数 $1/\lambda^p_1, \ldots, 1/\lambda^p_d$ で与えられる．よって $-\ln(\det(\boldsymbol{\Sigma}_p)) = -\ln\prod_{i=1}^d 1/\lambda^p_i = \sum_{i=1}^d \ln\lambda_i \ge d\ln\lambda^p_{\min}$ となる．次に，$\boldsymbol{\Sigma}_p$ の最小固有値が $1/\lambda^p_{\max}$ であるため，$\mathrm{tr}(\boldsymbol{\Sigma}_q^{-1}\boldsymbol{\Sigma}_p) \ge 1/\lambda^p_{\max}\mathrm{tr}(\boldsymbol{\Sigma}_q)$ となる． □

$\underline{\mathcal{R}}^q_{\mathrm{KL2}}(p)$ は $\lambda^p_{\min}, 1/\lambda^p_{\max}, (\boldsymbol{\mu}_q - \boldsymbol{\mu}_p)^t\boldsymbol{\Sigma}_q^{-1}(\boldsymbol{\mu}_q - \boldsymbol{\mu}_p)$ のそれぞれに対して単調増加するため，タイプ 1 の場合と同様のアルゴリズムで処理できる．なお，二次形式の距離 $(\boldsymbol{\mu}_q - \boldsymbol{\mu}_p)^t\boldsymbol{\Sigma}_q^{-1}(\boldsymbol{\mu}_q - \boldsymbol{\mu}_p)$ による最近傍問合せは，[12] の手法を用いることで，R-木などの空間索引を用いて処理できる．

以上，一般の場合に対する問合せ処理のアイデアについて述べた．近似を用いることで，問合せ処理の過程で実際に結果には含まれないオブジェクトに多数アクセスすることになるが，実際にどの程度になるかは実験で検証する必要がある．問合せ処理時間には，対象とするデータ集合の特性が大きく影響すると考えられる．個々のガウス分布位置がある程度離れているデータと密集しているデータで，戦略を変える必要があるかもしれない．

6. 関連研究

本研究では，KL 情報量に基づくガウス分布の正確な類似問合せ手法を提案した．つまり，問合せ処理の過程で近似処理は用いても，得られた top-k オブジェクトは真の top-k である．このような観点でのガウス分布の類似問合せは，著者の知る限り他に見られない．

KL 情報量は，対称でなく，また，三角不等式も満たさないため，nonmetric な尺度である．nonmetric な尺度を用いた類似検索に関するサーベイが [13] にある．検索処理の効率化のためのアプローチとしては，一つには，その尺度の特性を分析し，その尺度に応じた問合せ処理方式を開発するものがあり，本研究はこれに相当する．別のアプローチとしては，コストの高い前処理を行い，データ集合をうまく分離・整理して既存の索引構造を使えるようにするものもある [6], [13]．後者のアプローチに比べると，本稿での提案手法の前処理のコストは小さく，また，R-木など既存の空間索引をそのまま活用できるという利点がある．

KL 情報量やユークリッド距離など，さまざまな類似尺度を包含する尺度として *Bregman divergence* がある．より一般的な Bregman divergence について各種アルゴリズムを開発しようという流れもあり，たとえばクラスタリングに関する研究が [1] にある．[15] は，Bregman divergence による問合せのための効率的なアルゴリズムを開発しようというもので，目的は本研究に近い．ただし，彼らの研究では一次元の確率分布（ただし，d 個のヒストグラムにより表現されている）を扱っており，本研究の目的には使用できない．また，問合せの種類としてはタイプ 2 のみが考えられている．

[3] では，多次元ガウス分布の類似問合せが扱われており，目的は本研究との関連も深い．ただし，対象となるのは次元独立のガウス分布であり，類似度としてはガウス分布の積の積分が用いられている．

$$\int_{-\infty}^{\infty} f(\boldsymbol{x}) \cdot g(\boldsymbol{x}) d\boldsymbol{x} \tag{17}$$

ただし，$f(\boldsymbol{x}), g(\boldsymbol{x})$ は d 次元で各次元が独立なガウス分布である．なお，$f(\boldsymbol{x})$ を固定したとき，上記の式を最大にする $g(\boldsymbol{x})$ は $g(\boldsymbol{x}) = f(\boldsymbol{x})$ ではない[*2]．その点でこの尺度は「類似度」ではない．[3] では，この尺度に基づく類似問合せのために Gauss-木と呼ばれる索引を提案している．

ここでは，[3] で述べられた問合せが，本研究のアプローチで処理できることを簡単に述べておく．まず，1 次元の場合のランク付け関数は

$$\mathcal{R}^q_{\mathrm{PG}}(p) = \ln(\sigma_q^2 + \sigma_p^2) + \frac{(\mu_q - \mu_p)^2}{(\mu_q^2 + \mu_p^2)} \tag{18}$$

[*2] 1 次元の場合，$f(x) = \mathcal{N}(\mu_f, \sigma_f), g(x) = \mathcal{N}(\mu_g, \sigma_g)$ とすると，$f(x)$ を固定したとき上式を最大とする $g(x)$ は $\mu_g = \mu_f$ かつ $\mu_g \to 0$ のときに得られる．

となる．なお PG は “Product of Gaussian” を意味する．この関数は，σ_p^2 について，$\sigma_q^2 \geq (\mu_q - \mu_p)^2$ のとき単調増加し，そうでないとき，$\sigma_p^2 \leq (\mu_q - \mu_p)^2 - \sigma_q^2$ において単調減少，それ以外で単調増加する．KL 情報量の場合と異なり 3 つの領域に分かれるが，区分的に単調性を有する．

7. まとめと今後の課題

本稿では，KL 情報量に基づくガウス分布の類似問合せについてのアイデアを述べた．今後はアルゴリズムの洗練，実装手法の開発，評価実験を行いたい．

謝辞

本研究の経費の一部は内閣府最先端研究開発プロジェクト（FIRST）による．

参考文献

[1] Banerjee, A., Merugu, S., Dhillon, I. S. and Ghosh, J.: Clustering with Bregman Divergences, *J. of Machine Learning Research*, Vol. 6, pp. 1705–1749 (2005).
[2] Bishop, C. M.: *Pattern Recognition and Machine Learning*, Springer (2006).
[3] Böhm, C., Pryakhin, A. and Schubert, M.: The Gauss-Tree: Efficient Object Identification in Databases of Probabilistic Feature Vectors, *Proc. ICDE* (2006).
[4] Börzsönyi, S., Kossmann, D. and Stocker, K.: The Skyline Operator, *Proc. ICDE*, pp. 421–430 (2001).
[5] Chávez, E., Navarro, G., Baeza-Yates, R. and Marroquin, J. L.: Searching in Metric Spaces, *ACM Comput. Surv.*, Vol. 33, No. 3, pp. 273–321 (2001).
[6] Chen, L. and Lian, X.: Efficient Similarity Search in Nonmetric Spaces with Local Constant Embedding, *IEEE TKDE*, Vol. 20, No. 3, pp. 321–336 (2008).
[7] Cover, T. M. and Thomas, J. A.: *Elements of Information Theory*, John Wiley and Sons, 2nd edition (2006).
[8] Fagin, R., Lotem, A. and Naor, M.: Optimal Aggregation Algorithms for Middleware, *Proc. ACM PODS*, pp. 102–113 (2001).
[9] Fang, Y., Loparo, K. A. and Feng, X.: Inequalities for the Trace of Matrix Product, *IEEE Trans. on Automatic Control*, Vol. 39, No. 12, pp. 2489–2490 (1994).
[10] Ilyas, I. F., Beskales, G. and Soliman, M. A.: A Survey of Top-k Query Processing Techniques in Relational Database Systems, *ACM Comput. Surv.*, Vol. 40, No. 4 (2008).
[11] Ishikawa, Y., Iijima, Y. and Yu, J. X.: Spatial Range Querying for Gaussian-Based Imprecise Query Objects, *Proc. ICDE*, pp. 676–687 (2009).
[12] Seidl, T. and Kriegel, H.-P.: Efficient User-Adaptable Similarity Search in Large Multimedia Databases, *Proc. VLDB*, pp. 506–515 (1997).
[13] Skopal, T. and Bustos, B.: On Nonmetric Similarity Search Problems in Complex Domains, *ACM Comput. Surv.*, Vol. 43, No. 4 (2011).
[14] Zezula, P., Amato, G., Dohnal, V. and Batko, M.: *Similarity Search: The Metric Space Approach*, Springer (2006).
[15] Zhang, Z., Ooi, B. C., Parthasarathy, S. and Tung, A. K. H.: Similarity Search on Bregman Divergence: Towards Non-metric Indexing, *Proc. of VLDB Endowment (PVLDB)*, Vol. 2, No. 1, pp. 13–24 (2009).