動物移動行動データからの高速な頻出共起ルール抽出に関する検討

田　一鳴† 前川　卓也† 天方　大地† 原　隆浩† 松本　祥子††

依田　憲†† 藤岡　慧明††† 濱井　郁弥†††† 福井　大††††† 飛龍志津子††††††

† 大阪大学大学院情報科学研究科　〒 565-0871 大阪府吹田市山田丘 1-5

†† 名古屋大学大学院環境学研究科　〒 464-0814 愛知県名古屋市 千種区不老町 D2-2

††† 同志社大学 研究開発推進機構　〒 610-0394 京都府京田辺市多々羅都谷 1-3

†††† 同志社大学大学院生命医科学研究科　〒 610-0394 京都府京田辺市多々羅都谷 1-1

††††† 東京大学大学院農学生命科学研究科　〒 113-8657 東京都文京区弥生 1 丁目 1-1

†††††† 同志社大学生命医科学部　〒 610-0394 京都府京田辺市多々羅都谷 1-3

E-mail: †{tian.yiming,maekawa,amagata.daichi,hara}@ist.osaka-u.ac.jp, ††{s.matsumoto,yoda.ken}@nagoya-u.jp, †††{efujioka,shiryu}@mail.doshisha.ac.jp, ††††ctub1009@mail4.doshisha.ac.jp, †††††fukuidai@uf.a.u-tokyo.ac.jp

あらまし　本稿では，移動体から得られたマルチモーダル時系列データから高速に頻出ルールを抽出する手法の提案を行う．近年，センサ技術の進展により小型センサデバイスを動物に添付し，GPS により動物の移動軌跡を計測するとともに，環境・生体センサにより移動にまつわる時系列センサデータも同時に計測することが可能となりつつある．本研究では，移動およびセンサの時系列データを，それぞれ複数のモードにセグメントし，異なるモーダルを跨いだ頻出ルールを高速に抽出する手法を提案する．例えば移動軌跡から抽出された「局所探索」モードと生体センサから抽出された「緊張」モードが共起することが多いといったルールを抽出する．実データを用いた実験の結果から，提案手法は高速にルールを抽出できることを確認した．

キーワード　軌跡データ，セグメンテーション，頻出ルール

1. はじめに

センサデバイスの小型化やバッテリ技術の進展により，センサノードを移動体に添付して移動行動のセンシングを行う研究が盛んに行われている．特にユビキタスコンピューティング分野では人や車にセンサノードを添付して移動行動を観測する研究が多く行われてきた [1], [2]．近年のさらなるノードの小型化により海鳥などの動物にノードを添付し，移動行動を観測する手法の研究がユビキタスコンピューティングの研究コミュニティでも行われつつある [3]．

動物の行動情報をセンサデバイスを用いて記録する手法はバイオロギングと呼ばれ [4]，位置座標（軌跡）のみならず，環境データや生体データなどの様々な時系列データをセンサにより同時に計測することで，多くの研究者が動物の移動メカニズムの解明を目指している．すなわち，動物のデータの移動の解析には，軌跡情報のみでなく，同時に計測された時系列センサデータを同時に解析して，知識抽出を行う必要がある．

一般的に動物の移動は複数のモードから構成される．例えば，餌などの探索を局所的に行う「局所探索」や，餌場間などの移動を行う「長距離移動」などから構成されている．一方で，環境センサや生体センサから得られたデータにもモードは存在し，例えば脳波の時系列データから，動物の状態を「リラックス」モードと「緊張」モードに分けることもできる．動物から得られた複数のモーダルの時系列センサデータ（軌跡，環境データ，生体データなど）をそれぞれこのようなモードにセグメントし，異なるモーダルを跨いだ頻出パターンを抽出することで，生態学者にとって異なるモード間の相互関係を理解しやすい有用な知見を提供できる．例えば，ある動物が「長距離移動」モードにあるときは,「リラックス」モードであることが多いといった頻出ルールが考えられる．

このような，時系列データ（軌跡データやセンサデータ）からのモードの抽出には，系列データクラスタリング手法が用いられてきた．しかし，これまでの手法では，クラスタリングに用いる特徴量や，閾値などのパラメータは，研究者の先験的な知見から主観的に決められていた．そこで，本研究ではそのような特徴量や閾値を自動的に決定し，頻出ルールの抽出を行う手法を開発する．ここで，本研究が対象とする動物のデータからの頻出ルール抽出の最終目標は，有用度の高い頻出ルールを見つけることであるため，得られる頻出ルールの有用度が最も高くなるようなパラメータを同時に発見する（有用度の定義は後述する）．すなわち，それぞれの系列データをある特徴量とクラスタリングパラメータを用いてクラスタリングし，それらの結果から頻出ルールの抽出およびその有用度の計算を行う．この手順を，利用する特徴量とパラメータを変えながら何度も行い，高い有用度が得られたモード分けを，尤もらしいモード分けであると考える．ここで，時系列データをどの特徴とパラ

メータを使ってクラスタリングすればよいかには正解はない．しかし，何らかの特徴とパラメータにより得られた 2 つのモードが非常に頻繁に共起している場合，そのモード分けには何らかの意味があるはずであり，その結果を生態学者に提示することで，生態学的な知見が得られることが期待される．

しかし，軌跡データや時系列データから抽出される特徴量は数多く存在し，クラスタリングパラメータも様々な値を取り得る．そのため，時系列データのクラスタリングを何度も繰り返し行う必要があり，その計算量は膨大となる．そこで本研究では，パラメータを変更して何度も繰り返すクラスタリング処理を高速化して頻出ルールを導出する手法を提案する．時系列データのクラスタリングでは，時間的にスムーズなクラスタ化を実現するため，(1) まず時間的に近傍かつ値の類似しているデータポイント同士をグループ化（セグメント化）した後，(2) 時間的に離れたグループ同士を何らかの距離指標に基づいて併合することでクラスタリングを行うことが多い．すなわち，系列データのクラスタリングが 2 つのフェーズに分けられることに着目し，前半のフェーズを，前半のフェーズで利用されるパラメータ値を用いて計算したあと，その結果を用いて後半のフェーズを，後半のフェーズで利用されるパラメータ値を変更して何度も繰り返すことで，計算の高速化を実現する．

また，ある値のパラメータを用いて前半のフェーズの計算を行う際，他の値のパラメータで事前に計算された結果を再利用することで，高速化を行う．前半のフェーズでは，センサデータのシンボル化・離散化を行った後にセグメント化を行うが，大きなシンボル数で計算した離散化の結果は，小さなシンボル数での離散化の計算に再利用できることに着目する．さらに，後半のフェーズでは各セグメントをノードとするグラフのスペクトラルクラスタリング [7] を行う．その際に離散化された同じ値を持つノード同士の距離が 0 であることに着目して，元のグラフを同様のクラスタリング結果が得られる小さなグラフに変換することで，その計算時間を削減する．実データを用いた実験から，提案手法は比較手法に比べて大幅に計算時間を削減していることを確認した．

本論文の構成． 以下では，2. 章で関連研究について述べ，3. 章で提案手法を説明する．4. 章で性能評価のために行った実験の結果を紹介し，5. 章で本論文のまとめを行う．

2. 関 連 研 究

時系列クラスタリングに関する研究，および複数の時系列データ間の相関を計算する研究について紹介する．

時系列クラスタリング問題． 時系列クラスタリングは多くのアプリケーションに利用される基本的な処理であるため，これまでに高速化に焦点を当てた研究が数多く行われている [14]．時系列データは高次元データとみなせる．時系列クラスタリングでは，データ間の類似度（または距離）を計算する必要があるが，高次元データにおける類似度の計算コストは無視できない大きさである．そのため，データ間の類似度の計算回数を削減することが高速化を実現するために重要である．文献 [13] では，ランダムサンプリングを利用し，クラスタの中心を高速に決定するアルゴリズムを提案している．文献 [12], [15] では，相互相関と k-means 法に基づいたクラスタリングアルゴリズムを提案しており，Dynamic time warping 等の距離指標よりも正確かつ高速にクラスタリングできることを示している．1 つの時系列データにおける部分シーケンスをクラスタリングする問題では，モチーフ [9] よるクラスタリング [11] や，動径分布関数を利用してクラスタを決定するアルゴリズム [10] が提案されている．

以上の研究は，時系列データのクラスタリング自体を目的としているが，本研究では，有用度が高い頻出ルールを導くことを主目的としている．その目的を実現するため，パラメータを変えてクラスタリングが繰り返し実行される際の計算時間削減手法を提案する．

時系列データ間の相関計算問題． 1 つのデータからは特徴が抽出できないが，複数のデータ間で共起する特徴が存在する場合があり，これに焦点を当てた研究が行われている．文献 [16] は，ユーザが指定した条件（例えば，パターンの出現するデータの個数）に該当するパターンを抽出する問題に取り組んでいる．文献 [17], [18] では，異なるドメインのトランザクションの集合が入力された際，同時に出現するデータの組み合わせが閾値以上の回数で出現するものを抽出する問題に取り組んでいる．例えば，2 つの時系列データベース D_1 および D_2 が与えられ，それぞれあるタイミングでデータ o_1 および o_2 を持ったとする．もし $\langle o_1, o_2 \rangle$ がユーザの指定した閾値以上出現していれば，共起しているパターンとして抽出される．これらの研究では，パターンの頻出回数を効率的に計算することに焦点を当てており，本研究が行う有用度の高いルールが現れるパラメータを自動的に発見する問題とは異なる．

3. 提 案 手 法

概要． 図 1 に提案手法の概要を示す．提案手法は，移動軌跡（位置座標の時系列データ）と任意の次元数の時系列データ（環境センサデータなど）を入力とする．また，これらの時系列データは同時に記録されたものとする．これらの時系列データをそれぞれ，P と S とする．まず，それぞれの時系列データから特徴を抽出して，複数の 1 次元の特徴時系列データを計算する．P からは $P_0, P_1, ..., P_N$ の特徴時系列データが，S からは，$S_0, S_1, ..., S_M$ が得られる．（ただし，S に含まれる 1 つの次元のデータのみを抽出して S から抽出される特徴時系列データとしてもよい．）そして，それぞれの P_n と S_m のペアについて，ルールとそのスコアを導出する．スコア導出について説明する．P_n と S_m それぞれに対して様々なパラメータを用いて時系列クラスタリングを行うと，パラメータごとに時系列クラスタリング結果，$P_{n,\theta_0}, P_{n,\theta_1}, ..., P_{n,\theta_i}$ と $S_{m,\theta_0}, S_{m,\theta_1}, ..., S_{m,\theta_j}$ が得られる．P_{n,θ_i} と S_{m,θ_j} の組み合わせごとに，ルールとその有用度（スコア）のペア群を導出する．提案手法の擬似コードを Algorithm 1 に示す．

3.1 特 徴 抽 出

P からは，速度や角速度などの移動に関係する特徴量の時系

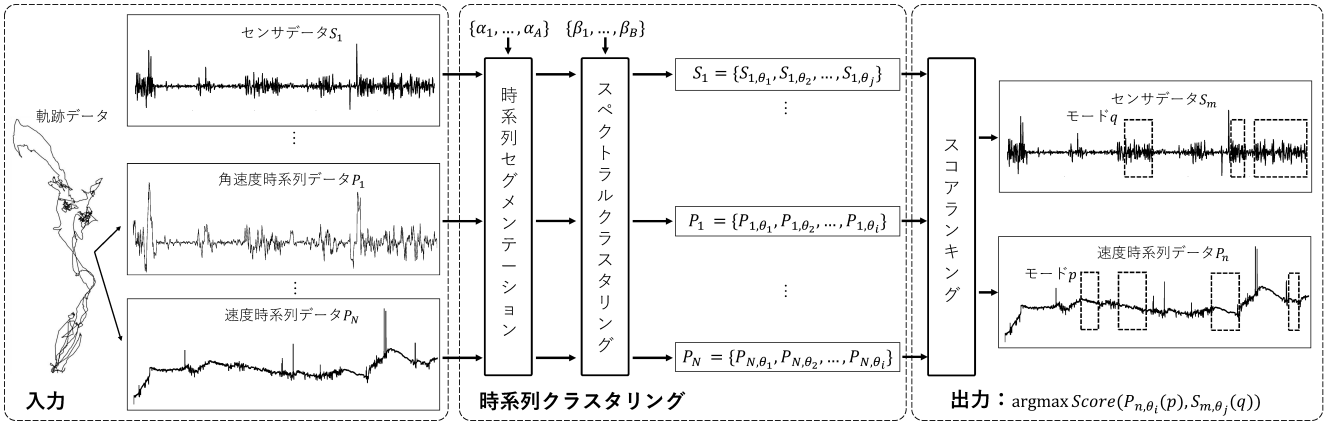

図 1　提案手法の概要

Algorithm 1: `Framework`

Input: T: a set of time-series with different attributes, α_{max}: maximum #symbols, α_{min}: minimum #symbols, β_{max}: maximum cluster size, β_{min}: minimum cluster size

```
/* Clustering time-series */
for ∀t_i ∈ T do
    μ, σ ← COMPUTE-MEAN-VARIANCE(s)  ▷ μ is mean and σ is variance
    for j = α_max to α_min do
        INCREMENTALSAX(t_i, μ, σ, j)
        MERGE-SYMBOLS(·)  ▷ merge the data with the same symbol into one segment
        SIMILARITY-COMPUTATION()  ▷ compute similarities between all segments
        V ← LAPLACIAN-EIGENMAPS()  ▷ V is a set of eigenvectors
        for k = β_min to β_max do
            C ← k-MEANS(V, k)  ▷ C is a set of clusters
            R ← R ∪ ⟨i, j, k, C⟩  ▷ store the clustering-result along with the related paramteres
/* Scoring clustering-result */
s* = 0, ⟨i, i'⟩ = ∅, j* = 0, k* = 0
for ∀r ∈ R do
    for ∀r' ∈ R\{r} s.t. r.j = r'.j ∧ r.k = r'.k do
        s ← max_{c∈r.C, c'∈r'.C} USEFULNESS(c, c')
        if s* < s then
            s* ← s, ⟨i, i'⟩ = ⟨r.i, r'.i⟩, j* = r.j, k* = r.k
Return s*, ⟨i, i'⟩, j*, k*
```

列データが導出される．S からは，そのセンサデータに応じた特徴量の時系列データが導出される．4.1 節にて，実験に用いたデータセットごとに利用した特徴を説明する．

3.2　時系列クラスタリング

得られた P_n（もしくは S_m）に対して，与えられたパラメータごとに網羅的に時系列クラスタリングを行う．時系列クラスタリングは，(1) 時系列セグメンテーションと (2) スペクトラルクラスタリングの 2 つの手順に分けられる．それぞれの手順においてパラメータが 1 つずつ存在 (シンボル数：α とクラスタ数：β) し，それぞれのパラメータの取る値が与えられている（$\alpha = [2, A], \beta = [3, B]$）．そして，パラメータの取る値のあらゆる組み合わせ $\theta_i = (\alpha_a, \beta_b)$ ごとにクラスタリングを行う．このとき，例えば (α_a, β_b) のパラメータの組を用いて時系列クラスタリングを行うとき，事前に (α_a, β_{b-1}) のパラメータの組を用いて計算を行っていた場合は，α_a を用いた時系列セグメンテーションの結果を再利用する．ただし，$\alpha_a > \beta_b$ とする．まず α_a を用いて時系列データを大まかにシンボル化・セグメント化し，さらに β_b を用いて詳細にクラスタ化を行うため，$\alpha_a > \beta_b$ を満たす必要がある．

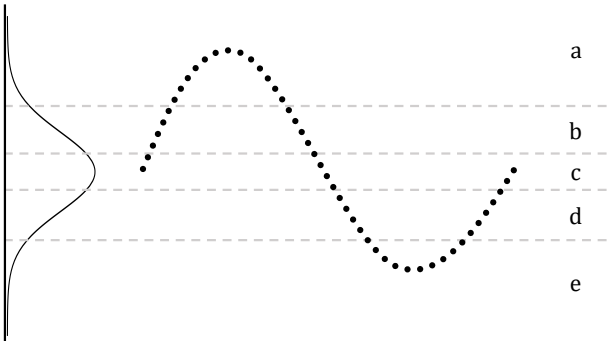

図 2　SAX の例．値域ごとに対応するシンボルを割り当て，それぞれのポイントをシンボル化する．

3.2.1　時系列セグメンテーション

ここでは，時間的かつ値的に近いデータポイントを 1 つのセグメントとして併合することで，(2) スペクトラルクラスタリングの計算時間を削減する．さらに，この処理の計算時間を削減するため，この手順を以下のように手順 1-1 と手順 1-2 に分ける．

手順 1-1. まず，P_n に含まれる全データポイントの平均と分散を計算する．この処理はパラメータ α_a を必要としないため，この結果は異なる α_a を用いてセグメンテーションするときも再利用する．

手順 1-2. SAX [6] を用いて各データポイントのシンボル化を行

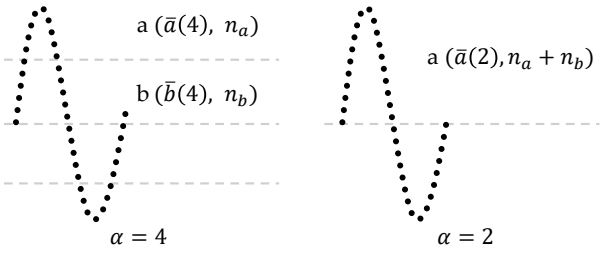

図 3　代表値の集約の例．$a = 2$ のときのシンボル a の代表値は，$\alpha = 4$ のときのシンボル a と b の結果から計算可能．

う．SAX では，図 2 に示すように，データ集合が正規分布に従うと考え，α_a 個のシンボルにシンボル化する場合，その正規分布の面積を α_a 等分にするよう $\alpha_a - 1$ 個の区切り点を定める．区切り点を定める際に，手順 1-1 で計算した全データポイントの平均と分散を再利用する．そして，それぞれのデータポイントがどのシンボルに対応する領域に入るかを判定することで，シンボル化を行う．この処理にかかる処理を削減するため，他の α_a を用いて計算された結果を再利用する．提案手法では，α の取りうる値を降順に並べ，大きい α の値から計算を行う．例えば図 3 に示すように，$\alpha = 4$ のときのシンボル a もしくは a に割り当てられたデータポイントは，$\alpha = 2$ のときは必ずシンボル a に割り当てられるため，$\alpha = 2$ のときはどのデータポイントがどのシンボルに対応するかの処理を省略できる．

次に，(2) スペクトラルクラスタリングで必要となる，各シンボルの代表値（平均）を求める．すなわち，あるシンボルに割り当てられたデータポイントの値の平均を求める．この代表値の計算にかかる処理を削減するため，他の α_a を用いて計算された結果を再利用する．例えば図 3 に示すように，$\alpha = 2$ のときのシンボル a の代表値は，$\alpha = 4$ のときのシンボル a と b の代表値とそれぞれのシンボルに割り当てられたデータポイント数（n_a,n_b）を用いて下記の通り計算できる．すなわち，$\alpha = 2$ のときのシンボル a に割り当てられた全データを用いる必要がない．

$$\bar{a}(2) = \frac{n_a \bar{a}(4) + n_b \bar{b}(4)}{n_a + n_b}$$

ただし，$\bar{a}(2)$ は，$\alpha = 2$ のときのシンボル a に対応するデータポイントの平均である．

最後に，時間的に隣接する同じシンボルをもつデータポイントをセグメントと考え，併合する．これにより，次の手順において扱うデータポイントの数を削減できる．

3.2.2　スペクトラルクラスタリング

次に，(1) 時系列セグメンテーションの結果のクラスタリングを，グラフ分割手法の 1 つであるスペクトラルクラスタリング [7] を用いて行う．ここでは，一つのセグメントをグラフの 1 つのノードとし，ノード（セグメント）間のエッジ（ノード間の類似度）の重みは，そのセグメントに対応するシンボルの代表値間の差の絶対値の逆数とする．例えば，シンボル a と b に対応するノード間のエッジの重みは下記の通りである．

$$w_{a,b} = \frac{1}{|\bar{a}(\alpha_a) - \bar{b}(\alpha_a)|} \tag{1}$$

そして，スペクトラルクラスタリングでは，カットするエッジの重みが最小となるように，グラフを β_b のサブグラフに分割する．ただし，長時間の時系列データの場合，ノード数が膨大となるため，グラフのノードの削減を行うことでスペクトラルクラスタリングの計算時間を削減する．また，さらに計算時間を削減するため，この手順を以下のように手順 2-1 と手順 2-2 に分ける．

手順 2-1.　まず，(1) 時系列セグメンテーションの結果からグラフを作成する．$P_{n,\alpha_a} = [s_0, s_1, ..., s_i, ...]$ を時系列セグメンテーションの結果とすると，s_i は i 番目のセグメントを表し，各セグメントがそれぞれノードとなるグラフを作成する．また，同じシンボルもしくは隣接するシンボル（例えば図 2 において，シンボル b に隣接するシンボルは a と c）をもつノード間にエッジを設定する．これは，後述するスペクトルクラスタリングの処理において，隣接しないシンボル同士が，間のシンボルをまたいで同じクラスタに割り当てることを防ぐためである（例えば，シンボル a と c に対応するノードが同じクラスタになり，b に対応するノードが別のクラスタに割り当てられるような状況）．i 番目と j 番目のセグメント間のエッジの重みは式（1）に定めた通りである．

次に，以降のスペクトラルクラスタリングにおける計算時間の削減を行うため，グラフのノード数の削減を行う．このとき，削減前のグラフと同様の結果が得られるようなノード数の削減を行う．スペクトラルクラスタリングでは，グラフ G を k 個のサブグラフ G_1, G_2,...,G_k に分割する．このとき，分割によりカットされるエッジの重みの和は下記の通り表される．

$$RatioCut(G_1, G_2, ..., G_k) = \sum_{i=1}^{k} \frac{Cut(G_i, \bar{G}_i)}{|G_i|} \tag{2}$$

$$Cut(G_i, G_j) = \sum_{n \in G_i, m \in G_j} w_{s(n),s(m)} \tag{3}$$

ただし，$s(n)$ は n 番目のノードに対応するシンボルである．式（2）を最小化するカットを見つけることで，類似したノードが同じクラスタになるようなクラスタリングを行うことができる．ここで，式（1）の通り，同じシンボルを持つノード同士の類似度は無限大となるため，同じシンボルを持つノードが異なるサブグラフ G_i と G_j に含まれる場合は，式（2）の値は ∞ となる．すなわち，$k < \alpha_a$（$k = \beta_b$）を満たす場合は同じシンボルのノードは同じサブグラフに属するため，同じシンボルのノード間のエッジがカットされることはない．そこで，提案手法では，同じシンボルをもつノードを 1 つに併合することで，ノード数の削減を行う．これにより，併合後のグラフにおける各ノード間の距離は下記のように表される．

$$W_{M_a,M_b} = \sum_{n \in S_a, n \in S_b} w_{s(n),s(m)} \tag{4}$$

ただし M_a をシンボル a をもつノードを併合したノード，S_a をシンボル a をもつノードの集合とする．このようなノードの併合を行った後でも，下式の通り，併合後のノード間のカットは併合前の同じシンボルをもつノードの集合間のカットと同じ

となるため，元のグラフと同じ分割結果が得られる．

$$Cut(S_a, S_b) = W_{M_a,M_b} = Cut(\{M_a\}, \{M_b\})$$

このノード数を削減したグラフを用いてスペクトラルクラスタリングを行う．スペクトラルクラスタリングでは，ノード間の類似度を示すラプラシアン行列を計算したあと，その行列の固有値分解を行うことで，固有ベクトルと固有値を求める．上記の処理は，パラメータ β_b を必要としないため，この結果は異なる β_b を用いてクラスタリングするときも再利用する．

手順 2-2. スペクトラルクラスタリングでは，上記手順で求めた固有ベクトルを基に，k-means++法 [8] を用いてノードのクラスタリングを行う．k-means++法のパラメータであるクラスタ数として β_b を用いる．

3.3 ルールとその有用度の導出

上記の手順で得られた P_{n,θ_i} と S_{m,θ_j} ごとに，ルールとそれに紐づく有用度（スコア）を計算する．

そして，すべての P_{n,θ_i} と S_{m,θ_j} の組み合わせから得られたスコアの高いルールをユーザに提示する．ここで，$P_{n,\theta_i}(p) = \{s_{p,1}, s_{p,2}, ...\}$ を P_{n,θ_i} から抽出した p 番目のモード（クラスタ）に属するセグメントの集合，$S_{m,\theta_j}(q) = \{s_{q,1}, s_{q,2}, ...\}$ を S_{m,θ_j} から抽出した q 番目のモードに属するセグメントの集合とする．上述した時系列クラスタリング手法はルールとスコアの算出方法に影響を受けるものではないが，本研究ではルールとスコアの算出方法を下記の通り定めた．

- Rule$(P_{n,\theta_i}(p), S_{m,\theta_j}(q))$: $P_{n,\theta_i}(p)$ は $S_{m,\theta_j}(q)$ とよく共起する．
- Score$(P_{n,\theta_i}(p), S_{m,\theta_j}(q))$ = Corr$(P_{n,\theta_i}(p), S_{m,\theta_j}(q)) \cdot$ inFreq$(P_{n,\theta_i}(p), S_{m,\theta_j}(q))$
- Corr $= \frac{\text{Overlap}(P_{n,\theta_i}(p), S_{m,\theta_j}(q))}{\sum_i |s_{q,i}|}$
- inFreq $= \frac{1}{\sum_i |s_{p,i}| + \sum_i |s_{q,i}|}$

ただし，$|s_{p,i}|$ はセグメント $s_{p,i}$ の長さとし，Overlap$(P_{n,\theta_i}(p), S_{m,\theta_j}(q))$ は $P_{n,\theta_i}(p)$ と $S_{m,\theta_j}(q)$ に含まれるセグメントがオーバーラップしている時間の長さとする．Corr は，p 番目のモードと q 番目のモードの共起度を表し，inFreq はほぼ定常的に起こっているモード同士に関するルールに対してペナルティを与えるという考えから，モードの出現時間の逆数としている．

ただし inFreq により，滅多に起こらないモードに関するスコアが高くなってしまうため，Corr $> \gamma$ となるルールのみを出力するものとする．ただし，L は軌跡全体のデータ長である．

4. 評価実験

本章では，提案手法の性能評価のために行った実験について説明する．

4.1 データセットと特徴量

提案手法の評価を，下記に示す 2 種類の動物の移動データを用いて行う．

海鳥. 新潟県岩船郡粟島に生息するオオミズナギドリ（*Calonectris leucomelas*）に取り付けたセンサデータロガー（Axy-Trek, Technosmart, Italy）から得られた GPS と水深センサデータを用いる．オオミズナギドリは，飛行中に海中へ飛び込むため，水深センサデータを取得している．これらのデータを提案手法を用いて分析することで，飛び込み行動と移動モードの関係性に関するルールが抽出されることが期待される．データ取得の詳細は [Matsumoto et al. 2017] に示す通りである．GPS データのサンプリング間隔は約 1 分であり，水深センサのサンプリングレートは 1.0Hz である．分析に用いた軌跡の数は 39 であり，その平均長は 364 時間，平均データ点数は 17,677 である．複数の軌跡を提案手法で分析するため，抽出した特徴時系列データを連結して提案手法に入力した．

移動軌跡からは，速度 [m/s]，加速度とその絶対値 [m/s^2]，角速度 [rad/s]，速度の移動平均・移動分散，加速度の移動平均・移動分散，加速度の絶対値の移動平均・移動分散，角速度の移動平均・移動分散を計算した．移動平均・分散の窓幅は 10, 20, 30, 40, 50 分に設定してそれぞれ計算した．すなわち，移動軌跡からは 33 の特徴時系列データを抽出した．クラスタリングパラメータである α は 20 から 4，β は 2 から 7 まで 1 きざみで変化させた．

水深センサデータは，1 分おきに平均を取ることで，GPS データのサンプリング間隔に合わせた．そして，そのデータに対して微分を計算したあと，絶対値を計算した．この処理により，オオミズナギドリが海に飛び込んだタイミングを表す時系列データが得られる．水深センサデータからは，1 つの特徴時系列データを抽出した．クラスタリングパラメータである α は 20 から 4 まで 1 きざみで変化させ，β は 2 のみとした．これは，水深センサデータからは海に飛び込んだタイミングとそれ以外の 2 つのモードが得られれば十分と考えたためである．また，γ は 60%とした．

コウモリ. 北海道大学苫小牧研究林において 16ch のマイクロホンアレイを用いて計測した採餌飛行中のモモジロコウモリ（学名 : Myotis macrodactylus）の 3 次元座標のデータを用いる．コウモリが採餌のために利用する池の周囲を 4 基のマイクロホンアレイで囲み，3 次元座標及び超音波の放射間隔を計測した．マイクロホンアレイによるデータ取得の詳細は [5] に示す通りである．コウモリが超音波を発した際，三辺測量の原理を用いてその 3 次元座標を計測する．その後、線形補間を用いてサンプリング間隔を一定に揃えた。また，コウモリは虫を捕食するタイミングで、通常の探索時よりも極めて短い間隔で超音波を発するため [5]，いつどこで捕食を行ったかを知ることができる．分析に用いた軌跡の数は 6 であり，その平均長は 67.97 秒，平均データ点数は 96000 である．

移動軌跡からは，速度 [m/s]，xy 平面の速度 [m/s]，速度の鉛直方向成分 [m/s]，加速度 [m/s^2]，xy 平面の加速度 [m/s^2]，加速度の鉛直方向成分 [m/s^2]，角速度 [rad/s]，速度の移動平均，xy 平面の速度の移動平均，加速度の鉛直方向成分の移動平均，加速度の移動平均，xy 平面の加速度の移動平均，加速度の鉛直方向成分の移動平均，角速度の移動平均を計算した．移動平均の窓幅は 5, 20, 30, 40 サンプルに設定してそれぞれ計算した．

表 1　オオミズナギドリのデータセットを用いた際の各手法の時系列クラスタリングに要した計算時間 (秒)．それぞれの手順の実行に要した時間も示す．

	手順 1-1	手順 1-2	手順 2-1	手順 2-2	合計
Proposed	0.17	4.12	1.30×10^{-2}	8.20	13.41
naive	2.80	13.81	3.02×10^{7}	18.50	3.02×10^{7}
w/o stats	2.84	4.32	1.40×10^{-2}	8.58	16.74
w/o symbol	0.16	6.57	1.40×10^{7}	8.42	16.09
w/o eigen	0.18	4.45	0.07	8.48	14.16
w/o reduce	0.17	4.22	3.31×10^{4}	16.18	3.32×10^{4}

すなわち，移動軌跡からは 35 の特徴時系列データを抽出した．クラスタリングパラメータである α は 20 から 4，β は 3 から 7 まで 1 きざみで変化させた．

移動軌跡と共に分析するセンサデータとしては，コウモリの捕食位置から算出した，コウモリの現在位置周辺の虫の密度の情報を用いた．捕食対象とする虫は短時間では群をなして一箇所に停滞しているとみなし，捕食位置から虫の空間密度を求めた．具体的には，コウモリの現在位置から半径 r メートル以内に存在する捕食位置の数を全ての捕食回数で割った値の時系列データを特徴時系列データとした．r は，0.5, 1.0, 1.5, 2.0 に設定してそれぞれ計算した．すなわち，4 の特徴時系列データを抽出した．クラスタリングパラメータである α は 20 から 4，β は 2 から 7 まで 1 きざみで変化させた．また，γ は 60%とした．

4.2 評価手法

本実験では，以下の手法を評価した．

- Proposed: 提案手法
- w/o stats: 手順 1-1 における正規分布の平均分散を毎回計算する手法
- w/o symbol: 手順 1-2 におけるシンボル化の結果を再利用しない手法
- w/o eigen: 手順 2-1 における固有値分解を毎回計算する手法
- w/o reduce: 手順 2-1 におけるグラフのノード数を削減しない手法
- naive: 手順 1-1 から手順 2-2 における計算量削減をすべて行わない手法

上の全ての手法の出力は同じであるため，本実験では計算時間による比較を行った．全ての手法は C++で実装されており，実験は，3.4GHz Intel Core i7 および 16GB RAM で構成される PC 上で行った．

4.3 評価結果

上記の手法をオオミズナギドリとモモジロコウモリのデータセットに適用した際の評価結果を表 1 と 2 に示す．ただし naive の計算時間に関しては，複数回のスペクトラルクラスタリングに長時間を要するため，1 回のスペクトラルクラスタリングに要する計測時間を基に算出した値を示している．表に示すとおり，オオミズナギドリのデータでは naive や w/o reduce は現実的な計算時間ではないことが分かる．データポイント数のノードから構成されるグラフのスペクトラルクラスタリングに非常に長時間を要する．w/o stats と比較すると，Proposed の手順 1-1 に要する時間は 1/10 以下となっている．また，w/o eigen と比較すると，Proposed の手順 2-1 に要する時間は 1/5 以下となっている．手順 1-1 および 2-1 に要する時間が自体が短いため，その影響も小さくなっているが，データポイント数が多いデータを解析する場合はその影響も大きくなる．w/o symbol と比較すると，Proposed の手順 1-2 に要する時間は 2/3 程度となっている．削減割合は小さいものの，SAX に要する計算時間が長いため，オオミズナギドリのデータでは約 2 秒ほどの削減となっている．

最後に，それぞれのデータセットから得られた最も有用性の高かったルールを紹介する．オオミズナギドリのデータセットから得られた最も有用性の高かったルールは,「加速度の絶対値の移動分散の時系列（$\alpha = 19, \beta = 3$，窓幅 $= 10$）」と「深度センサの微分の絶対値の時系列（$\alpha = 4, \beta = 2$）」のモード分けから得られたものであった.「加速度の絶対値の移動分散の時系列」から得られた,「加速度の絶対値の移動分散が大きいモード（約 $0.5(m/s^2)^2$ 以上）」と「深度センサの微分の絶対値が大きいモード（約 1Pa/s 以上）」がよく共起していた．Corr の値は 0.601，inFreq の値は 1/161857 であった．加速度の絶対値の移動分散から得られたモードは「速度を急激に変更するモード」，深度センサの微分の絶対値から得られたモードは「海面に飛び込むモード」と言うことができる．すなわち，オオミズナギドリが高速移動を行う「巡航モード」中に，餌となる魚群を発見し,「採餌モード」に移行するために速度を急激に落とした際に，海面に飛び込むことが多いという可能性を示唆している．図 4 に，それぞれのモードに対応するセグメントを赤色でハイライトした，ある個体から収集された軌跡データを示す．また，その際のそれぞれの時系列データも示す．赤色でハイライトされたデータ点が，上記のモードに対応するデータ点である．図に示すとおり,「速度を急激に変更するモード」は非常に短時間のモードであり,「海面に飛び込むモード」のうち約 1/3 がこのモードにおいて起こっていた．

モモジロコウモリのデータセットから得られた最も有用性の高かったルールは,「xy 平面の速度の時系列（$\alpha = 12, \beta = 3$」と「コウモリの現在位置周辺の虫の密度の時系列（$\alpha = 19, \beta = 2, r = 0.5m$）」のモード分けから得られたものであった.「速度が遅いモード（約 $3m/s$ 以下）」と「コウモリの現在位置周辺の虫の密度が高いモード（約 0.06 以上）」がよく共起していた．Corr の値は 0.601，inFreq の値は 1/232473 であった．すなわち，コウモリの周囲 0.5m 以内の虫の密度（すなわち過去と未来を通

表 2　モモジロコウモリのデータセットを用いた際の各手法の時系列クラスタリングに要した計算時間 (秒)．それぞれの手順の実行に要した時間も示す．

	手順 1-1	手順 1-2	手順 2-1	手順 2-2	合計
Proposed	1.60×10^{-3}	4.10×10^{-2}	6.50×10^{-3}	8.6×10^{-2}	0.14
naive	2.60×10^{-2}	1.40×10^{-1}	3.86×10^{3}	0.40	3.86×10^{3}
w/o stats	2.70×10^{-3}	4.70×10^{-2}	6.70×10^{-3}	8.7×10^{-2}	0.18
w/o symbol	1.60×10^{-3}	6.70×10^{-2}	0.70×10^{-3}	0.10	0.18
w/o eigen	1.60×10^{-3}	4.10×10^{-2}	3.10×10^{-2}	8.9×10^{-2}	0.17
w/o reduce	1.60×10^{-3}	5.20×10^{-2}	6.54×10^{2}	0.40	6.55×10^{2}

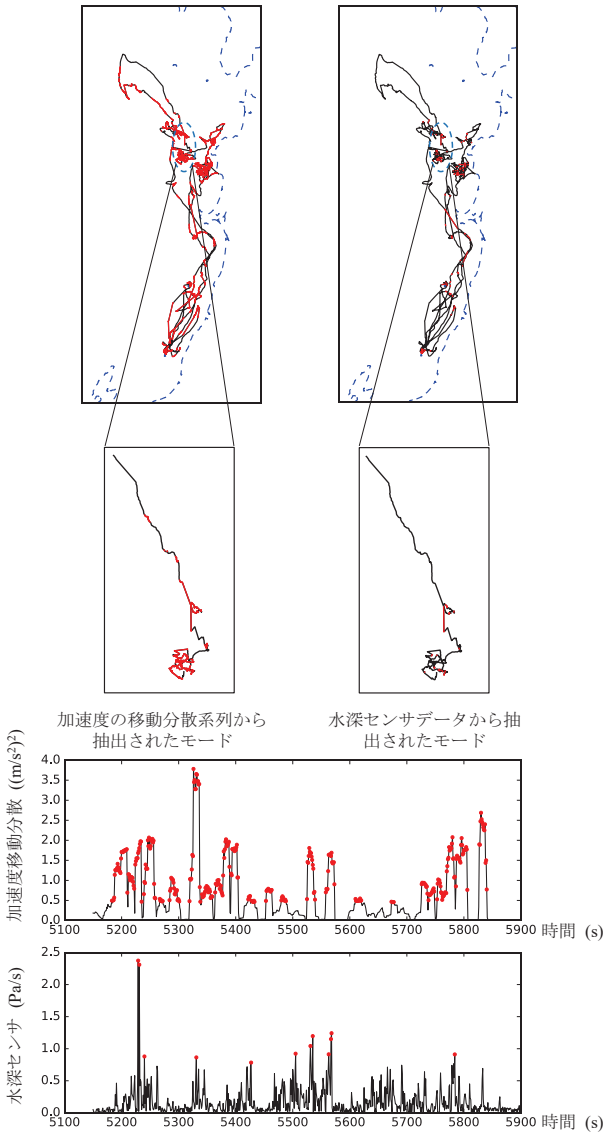

図 4　オオミズナギドリのデータから得られたルールの例

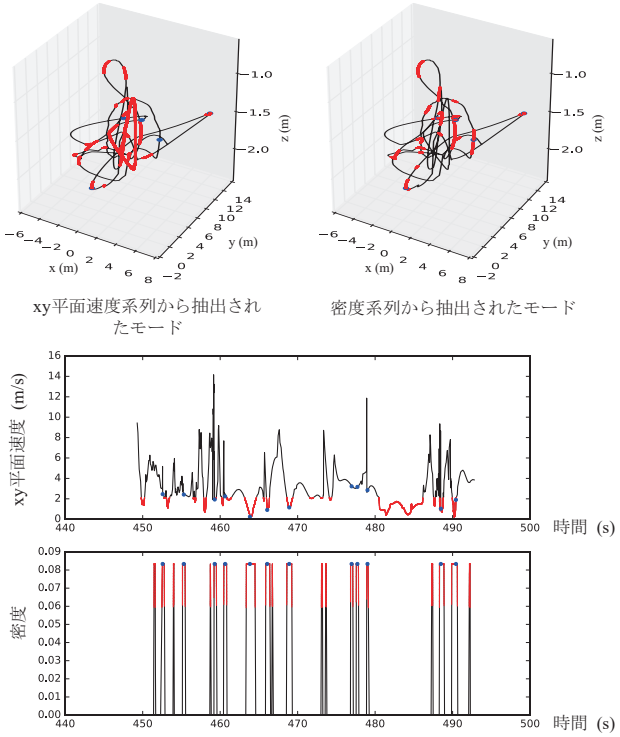

図 5　モモジロコウモリのデータから得られたルールの例

じて起こった捕食地点の密度）が高いとき，遅い速度で移動していた．図 5 に，それぞれのモードに対応するセグメントを赤色でハイライトした，ある個体から収集された 3 次元軌跡データを示す．また，その際のそれぞれの時系列データもグラフに示す．赤色のセグメントが抽出されたモードに対応し，青色のドットはコウモリが餌を捕食したタイミングを示す．密度の時系列データに示すとおり，コウモリは高速で移動しているため，0.5m 以内に捕食地点が含まれる時間は短く，密度の値が瞬間的に上昇している．また多くの場合，この瞬間的な上昇は捕食のタイミングに対応している．すなわち，餌にアプローチしているタイミングで速度が遅くなると言うことができ，コウモリが尾膜 (注1) で餌をすくい取る際に，速度に制限をかけている可能性が示唆された．

5. おわりに

本論文では，移動体から得られたマルチモーダル時系列データから高速に頻出ルールを抽出する手法の提案を行った．これまでの時系列データからのクラスタリングでは，主観的な特徴およびパラメータ選択が行われていたが，提案手法ではルール抽出と同士に特徴およびパラメータ選択も同時に行う．このとき，適切な特徴およびパラメータ選択を発見するため，高速にクラスタリングを繰り返しつつルール抽出を行う．評価実験で

(注1)：コウモリの尾と後ろ足の間の膜であり，餌の採取に用いられる．

は，提案手法を用いて海鳥およびコウモリから得られた軌跡データの分析を行い，計算時間削減を行わない単純な手法に比べて大幅な計算時間の削減を確認した．今後は，それぞれの動物から得られた頻出ルールの生物学的意味の詳細な検証を行う予定である．

6. 謝　　辞

本研究の一部は，JSPS 科研費 JP16H06539，JP16H06542，JP16H06541 の助成を受けて行われたものである。ここに記して謝意を表す．

文　　献

[1] Krumm, John and Brush, AJ Bernheim, “Learning time-based presence probabilities,” International Conference on Pervasive Computing, pp. 79–96, 2011.

[2] Yuan, Jing and Zheng, Yu and Zhang, Chengyang and Xie, Wenlei and Xie, Xing and Sun, Guangzhong and Huang, Yan, “T-drive: driving directions based on taxi trajectories,” SIGSPATIAL International conference on advances in geographic information systems (GIS), pp. 99–108, 2010.

[3] Alippi, Cesare and Ambrosini, Roberto and Longoni, Violetta and Cogliati, Dario and Roveri, Manuel, “A lightweight and energy-efficient Internet-of-birds tracking system,” IEEE International Conference on Pervasive Computing and Communications (PerCom), pp. 160–169, 2017.

[4] Rutz, Christian and Hays, Graeme C, ‘New frontiers in biologging science,” The Royal Society, 2009.

[5] Fujioka, Emyo and Aihara, Ikkyu and Watanabe, Shotaro and Sumiya, Miwa and Hiryu, Shizuko and Simmons, James A and Riquimaroux, Hiroshi and Watanabe, Yoshiaki, “Rapid shifts of sonar attention by Pipistrellus abramus during natural hunting for multiple prey,” The Journal of the Acoustical Society of America, Vol. 136, No. 6, pp. 3389–3400, 2014.

[6] Lin, Jessica and Keogh, Eamonn and Wei, Li and Lonardi, Stefano, “Experiencing SAX: a novel symbolic representation of time series,” Data Mining and knowledge discovery, Vol. 15, No. 2, pp. 107–144, 2007.

[7] Ng, Andrew Y and Jordan, Michael I and Weiss, Yair, “On spectral clustering: Analysis and an algorithm,” Advances in neural information processing systems, pp. 849–856, 2002.

[8] Arthur, David and Vassilvitskii, Sergei, “k-means++: The advantages of careful seeding,” ACM-SIAM symposium on Discrete algorithms (SODA), pp. 1027–1035, 2007.

[9] Li, Yuhong and Yiu, Man Lung and Gong, Zhiguo and others, “Quick-motif: An efficient and scalable framework for exact motif discovery,” IEEE International Conference on Data Engineering (ICDE), pp. 579–590, 2015.

[10] Denton, Anne M and Besemann, Christopher A and Dorr, Dietmar H, “Pattern-based time-series subsequence clustering using radial distribution functions,” Knowledge and Information Systems (KIS), Vol. 18, No. 1, pp. 1–27, 2009.

[11] Rakthanmanon, Thanawin and Keogh, Eamonn J and Lonardi, Stefano and Evans, Scott, “Time series epenthesis: Clustering time series streams requires ignoring some data,” IEEE International Conference on Data Mining (ICDM), pp. 547–556, 2011.

[12] Rakthanmanon, Thanawin and Keogh, Eamonn J and Lonardi, Stefano and Evans, Scott, “k-shape: Efficient and accurate clustering of time series,” ACM SIGMOD International Conference on Management of Data, pp. 1855–1870, 2015.

[13] Ding, Rui and Wang, Qiang and Dang, Yingnong and Fu, Qiang and Zhang, Haidong and Zhang, Dongmei, “Yading: Fast clustering of large-scale time series data,” PVLDB, Vol. 8, No. 5, pp. 473–484, 2015.

[14] Aghabozorgi, Saeed and Shirkhorshidi, Ali Seyed and Wah, Teh Ying, “Time-series clustering–A decade review,” Information Systems, Vol. 53, pp. 16–38, 2015.

[15] Paparrizos, John and Gravano, Luis, “Fast and Accurate Time-Series Clustering,” ACM Transactions on Database Systems (TODS), Vol. 42, No. 2, pp. 8, 2017.

[16] Zhu, Xingquan and Wu, Xindong, “Discovering relational patterns across multiple databases,” IEEE International Conference on Data Engineering (ICDE), pp. 726–735, 2007.

[17] Gwadera, Robert and Crestani, Fabio, “Discovering Significant Patterns in Multi-stream Sequences,” IEEE International Conference on Data Mining (ICDM), pp. 827–832, 2008.

[18] Peng, Wen-Chih and Liao, Zhung-Xun, “Mining sequential patterns across multiple sequence databases,” Data Knowledge Engineering (DKE), Vol. 68, No. 10, pp. 1014–1033, 1970.