時系列データベースからの知識発見における次元圧縮に関する検討

平　和幸† 　黒木　進‡ 　北上　始‡ 　森　康真‡

†広島市立大学情報科学部　〒731-3194 広島市安佐南区大塚東三丁目 4 番 1 号
‡広島市立大学大学院情報科学研究科 〒731-3194 広島市安佐南区大塚東三丁目 4 番 1 号
E-mail: †taira@de.info.hiroshima-cu.ac.jp, ‡{kuroki,kitakami,mori}@hiroshima-cu.ac.jp

あらまし　時系列データベースから知識発見する場合，時系列データベースを次元圧縮して，より単純なデータにしてから活用することが多い．ただし，次元圧縮を行うと時系列データベースに記憶されたデータ全体の構造が変化する．今回は FastMap を用いた次元圧縮について，データ全体の構造変化により得られる知識にどのような変化が生じるか，例を用いて検討する．

キーワード　時系列解析，次元圧縮

A Study on Dimension Reduction on Knowledge Discovery from Time Series Databases

Kazuyuki TAIRA† 　Susumu KUROKI‡ 　Hajime KITAKAMI‡ 　and 　Yasuma MORI‡

†Faculty of Information Sciences, Hiroshima City University.

3-4-1 Ozukahigashi, Asaminami, Hiroshima, 731-3194 Japan
‡Graduate School of Information Sciences, Hiroshima City University.

3-4-1 Ozukahigashi, Asaminami, Hiroshima, 731-3194 Japan
E-mail: †taira@de.info.hiroshima-cu.ac.jp, ‡{kuroki, kitakami,mori}@hiroshima-cu.ac.jp

Abstract　Time series databases are often mapped to a low dimensional space by dimension reduction method and then analyzed and utilized, when we want to find knowledge from the databases. But when the dimensions of the time series databases are reduced, whole structures of time series databases are often distorted. We study such kind of distortion and show how distortion affects knowledge discovered from time series databases using examples.

Keyword　Time Series Analysis，Dimension Reduction

1. はじめに

本研究は時系列データベースの次元圧縮に関して次元圧縮前と次元圧縮後のデータを比べることにより，圧縮する際にどのような変化が生じたかを検討していく．菅野による研究[2]により FastMap[1]を用いて時系列データの次元圧縮が行われている．今回はこの結果に対し CG による可視化を行う．可視化により得られた情報を利用してデータ全体の構造を把握し,視覚的に比較を行っていく．

2. FastMap

FastMap とは高次元のデータセットの次元数を圧縮する技術である．圧縮したい次元数を k とする．n 次元のデータセットの中から距離関数 D を用いて最も距離の離れたオブジェクトの対（O_a, O_b）を pivot object として選び直線 O_aO_b を k 次元の一本目の軸にとる．最も離れたオブジェクトを選ぶのは距離をよく保つためである．任意のオブジェクト O_i の一つ目の座標値 X_i は，O_i を直線 O_aO_b に投影し，O_a から投影した点 E までの距離が X_i となる．X_i は余弦定理により求めることができる．

残りの 2 本目から k 本目の軸は次のように決める．直線 O_aO_b に直交する n-1 次元の超平面に全てのオブジェクトを投影する．この超平面に投影されたオブジェクト間のユークリッドの距離を求めて距離関数 D を更新する．更新した距離関数 D’を用いて，一本目と同様に，二本目の軸とその座標値を求める．これを k 回繰り返すことによって，k 次元の座標値を得ることができる．

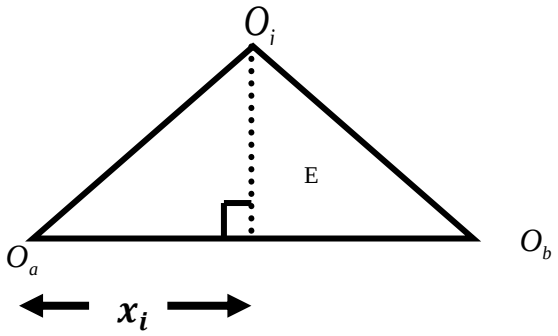

図 1：直線 O_aO_b への投影

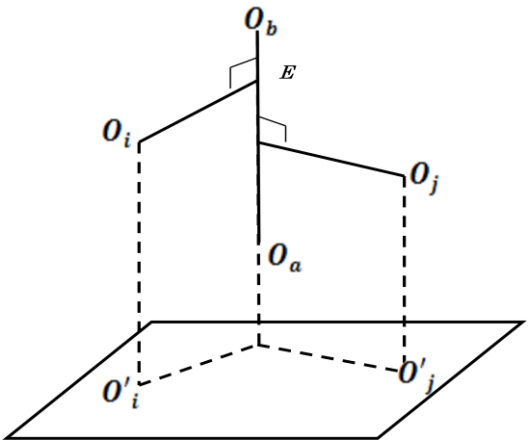

図 2：直線O_aO_bに垂直な超平面への投影

3. 時系列間の距離

FastMap は距離情報だけを必要とする．そこで本研究では菅野が使用した距離の内，フーリエ変換を用いたフィルタ出力に基づく距離に対し改良を施し使用した．

まず 2 つの時系列 A,B を最大値 1,最小値が 0 となるように正規化する．正規化した各時系列に離散フーリエ変換を施し，得られた周波数スペクトルの差を時系列間の距離とする．

時系列を離散フーリエ変換することにより，周波数成分を求めることができる．離散フーリエ変換は次の式によって行われる．

$$X_j = \sum_{k=0}^{n} x_k e^{-\frac{2\pi i}{n} jk}$$

(n:データのオブジェクト数，π:円周率，i:虚数単位)

求めた周波数成分のうち，第一成分と第二成分を取り出し距離を算出する．時系列 A と時系列 B の距離を求める場合は以下の式を用いる．時系列 A の(第 1 成分，第 2 成分)を$(\hat{A}_1, \hat{A}_2)$とする．

$$d(A,B) = \sqrt{\left|\hat{A}_1 - \hat{B}_1\right|^2 + \left|\hat{A}_2 - \hat{B}_2\right|^2}$$

求めた値を時系列間の距離とする．

4. FastMap による可視化

600 銘柄の 2007 年 7 月 17 日～2007 年 12 月 7 日までの 100 日分の株価データを使用し，FastMap により次元圧縮し可視化を行った．

類似関係がうまく表されているなら近くにマップされているオブジェクト同士の波形は似たような形で，遠くにマップされているオブジェクト同士の波形は対照的な形になるはずである．

2 次元の図では次のようになる．

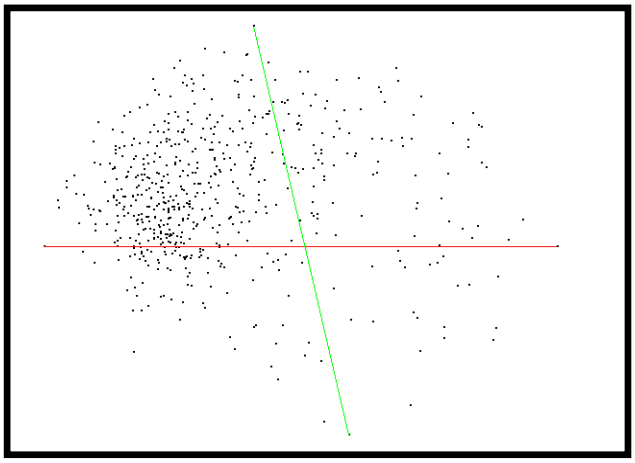

図 3：FastMap による可視化(2 次元)

図 3 において分布している点がそれぞれ時系列データとなっている．今回は 600 銘柄のデータを使用しているので，600 個の点が存在することになる．

本研究ではこの 2 次元の図から時系列データベースの圧縮後のデータ全体の構造を知るために,ボロノイ図を用いた領域分割と階層的クラスタリングを用いたクラスタリングを行った．

5. ボロノイ図

ボロノイ図とは，ある距離空間上の任意の位置に配置された複数個の点に対して，同一距離空間上の他の点がどの母点に近いかによって領域分割された図のことである．

ボロノイ図は，図 4 のように母点と定めた数だけ領域を分割できる．分割する境界線は各々の母点の垂直二等分線の一部となる．

よって今回作成した FastMap において母点を任意に設定することにより，母点の数だけ時系列データの分布を分類することができる．実際にかぶせてみると図 5 のようになる．

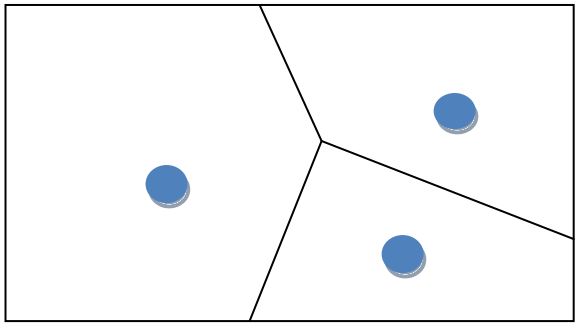

図 4：2 次元ボロノイ図

今回は母点の数は 15 としボロノイ図を描いた．母点として選んだ銘柄は，各業種の中で時価総額が最も高いものをそれぞれ選び,その中で上位 15 銘柄を母点として選んだ．

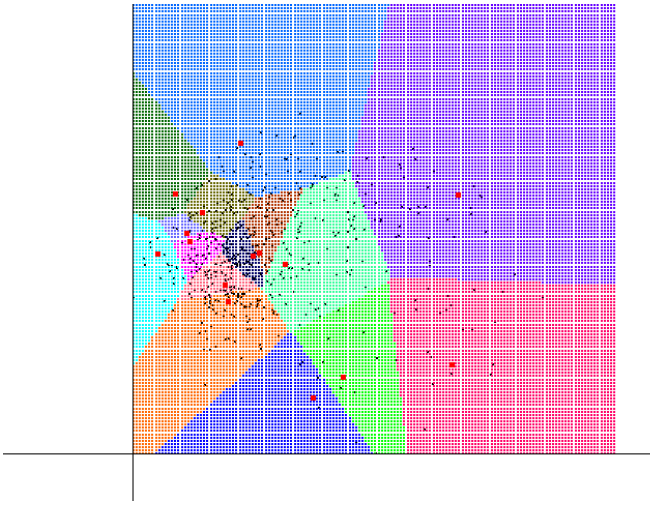

図 5：ボロノイ図をかぶせた 2 次元図(母点が赤丸)

6. 圧縮前との比較

図 5 で示したボロノイ図は時系列データベースを 2 次元に圧縮したもの使って描いている．この図に対し圧縮前の時系列データベースとの比較を行う．

ボロノイ図は母点との距離を算出し，どの母点に近いかで領域分割した図である．図 5 では 2 次元に圧縮したデータから母点との距離を算出したが，ここでは圧縮前のデータを用いどの母点に近いかで点を分類する．

図 6 は圧縮前の距離を使って母点ごとに領域分割した図である．各領域の点が圧縮前のデータを用いた分割でも同じ領域に分割されれば，領域の色と点の色が一致している．

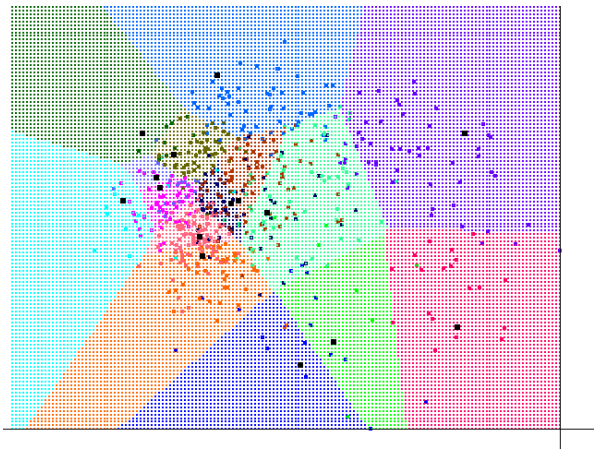

図 6：圧縮前の情報を元にした領域分割

また，圧縮後の分割が圧縮前の分割と比べてどのくらい違っているかを図を用いて検討する．

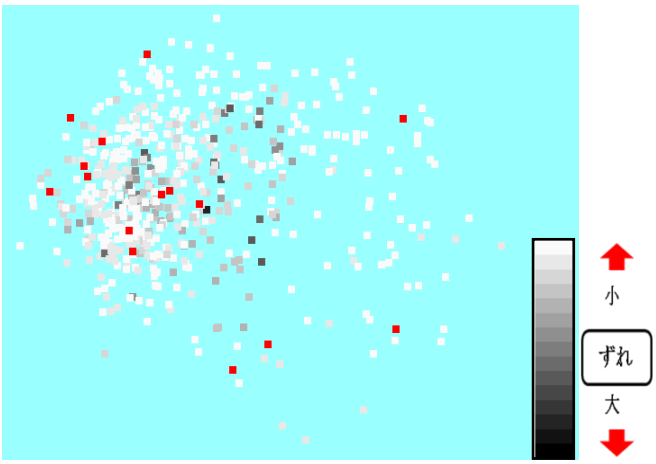

図 7：圧縮前との分割のずれ

図 7 は各点においてどの母点に近いかの順位を求め，2 次元図にしたときとどれぐらい違うかを図にした．点の色が黒に近ければ圧縮前とずれた分割になっており，点の色が白に近ければ圧縮前と近い分割になっている．他の母点と離れた母点の周辺の点は正しく分類される傾向がある．また，ボロノイ領域の境界付近や点の密集した領域では分類の食い違いが見られるが，大きく食い違う点は比較的少ない．

図 8 は FastMap を用いて次元圧縮する際に更新していく距離行列を用いて，更新前と比べてどのくらい変わったかを図にしている．FastMap で次元圧縮を行う場合，距離行列を更新しながら圧縮を行う．更新前の距離行列 D と更新後の距離行列 D''は次のようになる．

$$D=\begin{pmatrix} 0 & d_{ab} & d_{ac} & \cdots & d_{aj} \\ d_{ba} & 0 & d_{bc} & \cdots & d_{bj} \\ d_{ka} & d_{kb} & 0 & \cdots & d_{kj} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ d_{ja} & d_{jb} & d_{jk} & \cdots & 0 \end{pmatrix} \qquad D''=\begin{pmatrix} 0 & d''_{ac} & d''_{ak} & d''_{aj} \\ d''_{ca} & 0 & d''_{ck} & d''_{cj} \\ d''_{ka} & d''_{kc} & 0 & d''_{kj} \\ d''_{ja} & d''_{jc} & d''_{jk} & 0 \end{pmatrix}$$

更新された距離行列 D''の内，要素から値e_kを以下のようにして求める．

$$e_k = \left(d''_{ka}\right)^2 + \left(d''_{kc}\right)^2 + \cdots + \left(d''_{kj}\right)^2$$

更新前の距離行列 D の要素と更新後の距離行列 D''の要素から求めた e_k を使って k 番目の時系列から他の時系列への距離の平均残存度 f_k を求める．

$$f_k = \frac{e_k}{\left\{\left(d_{ka}\right)^2 + \left(d_{kc}\right)^2 + \cdots + \left(d_{kj}\right)^2\right\}}$$

求めた平均残存度が 1 に近いほど圧縮による減少が少ない．また平均残存度は圧縮前の距離と圧縮後の距離の差なので，値が小さいほど圧縮による影響が少ないと言える．平均残存度を図にしたものが図 8 である．

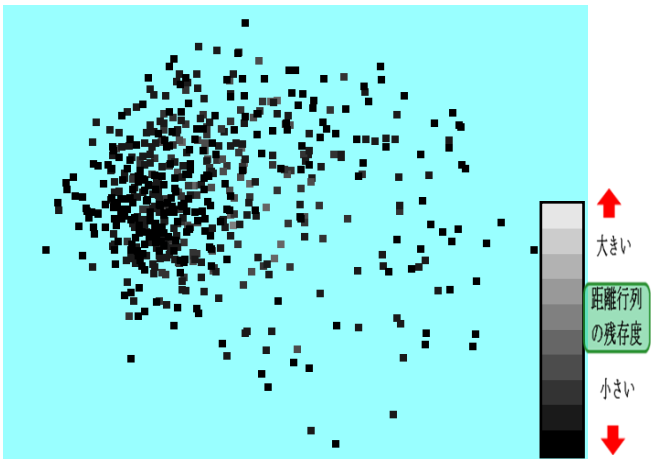

図 8：次元圧縮による距離の平均残存度

点の色が黒に近いほど，平均残存度が小さい．また，調べた結果全体の 92%の時系列が平均残存度が 0.3 以下であることが分かった．よって今回用いたデータは全体的に見て次元圧縮における距離行列の平均残存度が低いデータであることが分かった．平均残存度が低いことから今回用いたデータにおける FastMap による次元圧縮は，圧縮前の距離の情報をある程度保った次元圧縮になっていることが分かった．

7. 階層的クラスタリング

階層的クラスタリングとはクラスタリング手法の一つである．N 個のデータがある時，それぞれを別のクラスタとみなし N 個のクラスタがある状態を初期状態とする．各データ間の距離を利用しクラスタ間の距離を更新しながら，クラスタ間距離が短いものを統合していく．

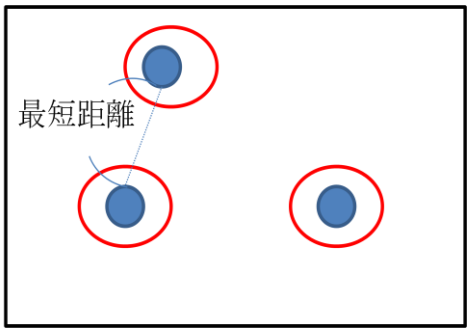

図 9：最短距離を見つける

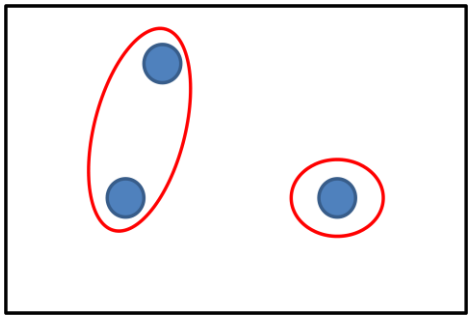

図 10：最短距離のものを統合する

今回クラスタ間距離として採用したのは最短距離である．この手法は 2 つのクラスタ間距離をそれぞれのクラスタに属するデータの組み合わせの中でデータ間の距離が最も短いものをクラスタ間距離とする．

この手法により，密集したものどうしをクラスタリングすることができる．

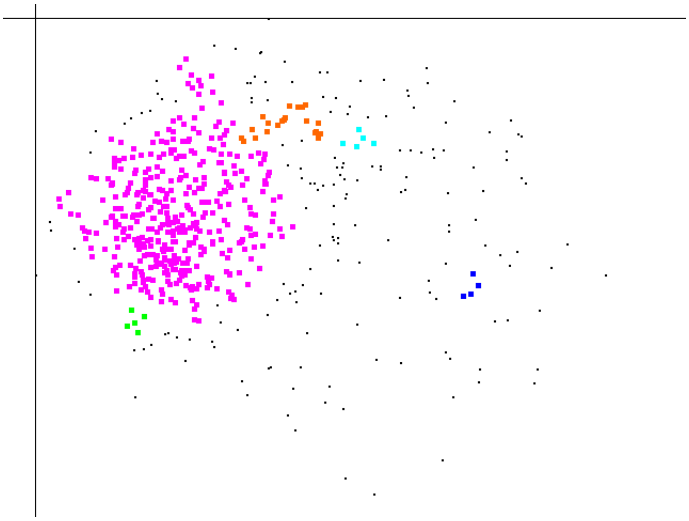

図 11：圧縮後の距離を用いた階層的クラスタリング

図 11 はクラスタ数を 100 とした場合の階層的クラスタリングで，要素数の多いクラスタ上位 5 つを色分け

している．大きなクラスタが 1 つできていることと，クラスタ内に境界線を引けそうであることが分かる．

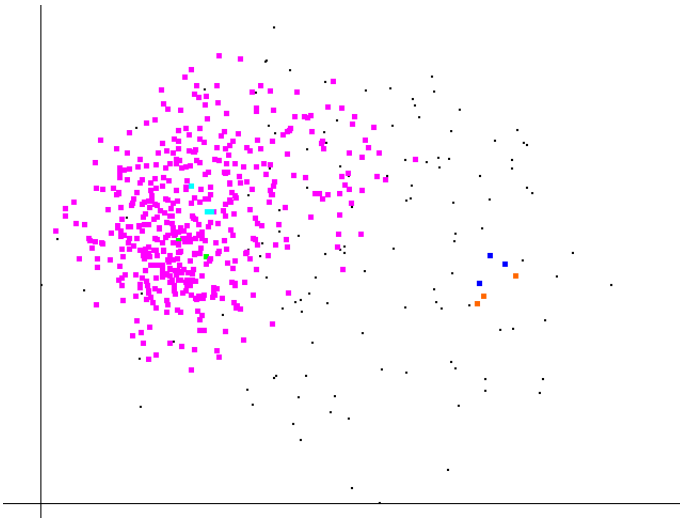

図 12：圧縮前の距離を用いた階層的クラスタリング

図 12 はクラスタ数を 100 とし，データ間の距離として圧縮前の時系列データ間の距離をそのまま用い，クラスタリングを行い，2 次元図にクラスタごとに色分けして描画した．圧縮前と同様，要素数が多いクラスタ上位 5 つを色分けしている．

次元圧縮前に最も要素数の多いクラスタは，次元圧縮後も要素数が最多である．しかし要素数は減少し，いくつかにわかれたことも分かる．また，次元圧縮後にクラスタリングを行っても，次元圧縮前に行ったクラスタリング結果とは大きな違いはない場合があることも分かった．そうなったのは図 8 に示したように今回とりあげた時系列データベースが次元圧縮する前と後で，距離の減少率が比較的小さいマッピングが選べる性質を持ったデータセットであったからだと考えられる．

8. おわりに

本研究ではボロノイ図やクラスタリングの効果が次元圧縮によりどのように変化するかを検討した．距離の近さに応じて分類するやり方の場合は，次元圧縮の影響を受けにくく，また，その影響は十分受け入れ可能なものであると考えられる．

謝辞

本研究の一部は，日本学術振興会，科学研究費補助金(基盤研究(C))，課題番号 ：20500137)の支援により行われた．

文　献

[1] C.Faloutsos, K.-I. Lin: FastMap: A Fast Algorithm for Indexing, Data-Mining and Visualization of Traditional and Multimedia Datasets, SIGMOD RECORD, Vol.24, No.2, pp. 163-174, June 1995.

[2] 菅野亮一: FastMap を用いた時系列の類似関係の可視化,広島市立大学情報科学部卒業論文, 2008.