データの傾向を考慮したストリームデータの相関の計算方法の提案

Calculating Correlations of Stream Data with Data Tendencies: A Proposal

畑中　洋介♥　　有次　正義♦

Yousuke HATANAKA　Masayoshi ARITSUGI

本稿では複数のストリームを監視し，ストリームのデータ傾向を考慮することによって誤差を少なくするストリーム相関の効率の良い求め方を提案する．ストリームデータの相関は時間によって変わるため，できるだけ効率よく相関を求めることが重要である．本稿では， sliding window を階層構造で管理することを考える．この際，データの傾向を考慮した近似を行うことにより，精度のよい相関の計算を可能にする．

A method for efficiently and accurately calculating correlations of stream data with taking into account of data tendencies by monitoring multiple stream data is proposed in this paper. Since correlations of stream data change with time, it is important to calculate them efficiently. In this paper, we try to manage a sliding window in a hierarchical structure. We show that taking into account of data tendencies allows us to obtain good approximation of correlations.

1. はじめに

ストリームデータを扱うアプリケーションにおいては，多少の誤差が生じても，より速くユーザからの問合せに答えることが重要であると考えられる．このことは Data Stream Management System[1]を設計する際の問題として挙げられている．ストリームデータは株式市場やセンサーモニター，Web サイト管理，ネットオークション，医療の分野において非常に多く発生している．ストリームデータは量が膨大で，新しいデータが極めて早いペースで到着するといった特徴がある．

本稿では複数のストリームデータを監視し，ストリームデータの相関を効率よく求める手法を提案する．相関を求める際，データの傾向を考慮することによって誤差を少なく近似して求めることを考える．ストリームデータの相関は時間によって変わるため，できるだけ効率よく相関を求めることが重要である．同時に，ストリームデータの平均，分散という統計値についても考慮する．ストリームデータの相関等を求めることによって例えば以下のような利点が考えられる．株式市場で発生しているストリームデータの相関を求めることができれば，連動している 2 つの銘柄を見つけることができ，ペアトレードをしている裁定業者にとって有利になる．分散を効率良く求めることができれば，ある時間内において激しく変動している銘柄を見つけることができる．センサーモニターで発生しているストリームデータの温度の相関を求めることによって，温度変化が連動している場所を特定することができ，1 つの場所で異常気象が発生した際もう 1 つの場所でも同じような異常気象が発生することが予想できる．降水量と川の水位の相関を求めることにより災害が発生しやすい場所を特定することができる．Web サイトを訪れたユーザの閲覧した軌跡を表すクリックストリームと呼ばれるストリームデータの相関を求めることにより，バナー広告，メールマガジンの効果が分かる．

ストリームデータの相関を求めるためには，相関係数を求めれば良い．相関係数を求めるために StatStream[2]で提案されている DFT 係数を使った近似手法を用いる．この手法を用いることによって，全てのデータを格納することなく，ストリームデータの相関係数を DFT 係数から近似して求めることができる．本稿では，StatStream で提案されている分割された sliding window を木構造で扱うことによって，最新のデータの誤差を小さく保ちながらストリームデータの平均，分散，相関という統計値を効率良く求める方法を提案する．統計値を求める際，ノード内のデータの傾向を考慮することによって上位レベルの誤差が少なくなるような DFT 係数更新方法を提案する．

2. 関連研究

関数の分解手法として wavelet がある．SWAT[3]では最も簡単な Haar wavelet を用いてストリームデータを近似的に扱っている．SWAT では最新のデータの誤差が小さくなるような構造をとっており，SWAT の木構造の各ノードにはデータの平均値が格納されている．SWAT ではシングルストリームデータの平均値のみを扱っている．一方 StatStream[2]では，相関係数というマルチストリームデータの統計値について扱っている．本稿では StatStream で提案されている分割された sliding window を SWAT のような階層構造で扱うことを考える．

2.1 Wavelet

Wavelet は関数を階層的に分解する手法である．Haar wavelet は最も単純な wavelet で，理解と実装が簡単だという特徴がある．1 次元データセット[2 8 3 3]を例として考える（表 1）．Wavelet 分解では初めにペアごとの平均値を計算する([5 3])．この処理で元のデータセットの情報が捨てられてしまう．しかしデータ値の差を 2 で割った[-3 0] (Detail Coefficient) を格納しておくことにより，データの平均値から元のデータを復元することが可能となる．この平均値計算と差を 2 で割る計算が繰り返し行われる．

表 1　Wavelet 分解

Table 1　Wavelet decomposition

Average	Detail Coefficient
[2 8 3 3]	
[5 3]	[-3 0]
[4]	[1]

♥ 学生会員　群馬大学大学院工学研究科博士前期課程
hatanaka@dbms.cs.gunma-u.ac.jp
♦ 正会員　群馬大学工学部情報工学科
aritsugi@cs.gunma-u.ac.jp

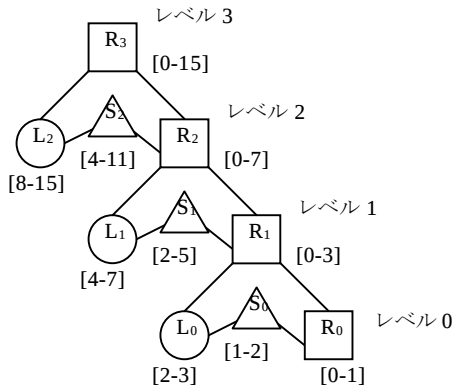

図 1 sliding window サイズが 16 の時の SWAT
Fig.1 SWAT in case the size of sliding window is 16

2.2 SWAT

SWAT(Stream Summarization using Wavelet-based Approximation Tree)[3]は Haar wavelet を用いてストリームデータを近似する．これにより元のデータを全て格納することなく，各ストリームデータの値を近似的に求めることができる．時系列ストリームデータではデータ値が連続しているため，Detail Coefficient は 0 または小さい値となる．よって Detail Coefficient を省略しても誤差は小さくなることを利用して，SWAT ではノードに平均値のみを格納している．例として sliding window のサイズが 16 の SWAT を図 1 に示す．各ノードにはインデックスに含まれるデータの平均値が格納されている．例えば R_1 のインデックスは[0-3]であり最新のデータから 3 番目までのデータの平均値が格納されている．つまり，SWAT では新着データの近似の精度が古いデータの精度より高くなっている．[3]によれば，SWAT は histogram による方法と比べると問合せ応答時間が早く，誤差が少ないという特徴がある．データの更新はノードのレベル毎に異なり，レベル i のノードは 2^i 個のデータが到着する毎に以下のように計算される．

- ノード L_i にノード S_i のデータをシフトする
- ノード S_i にノード R_i のデータをシフトする
- ノード R_i のデータはノード L_{i-1} とノード R_{i-1} から計算

本稿では SWAT の階層構造を応用して最新のデータの誤差が少なくなるように平均，分散，さらに相関係数を求めることを可能にする手法を提案する．

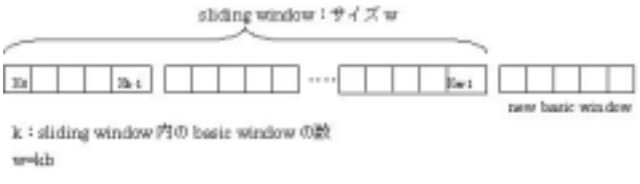

図 2 sliding window と basic window
Fig.2 A sliding window and basic windows

2.3 StatStream

StatStream(Statistical Monitoring of Thousands of Data Streams in Real Time)[2]では sliding window 中の平均，標準偏差，分散といったシングルストリームデータの統計値だけでなく，ストリームデータ同士の相関係数というマルチストリームデータの統計値についても考えている．図 2 のように， sliding window を同じ大きさの basic window に分け，basic window 毎に統計値を格納し全体の統計値を更新する．b 個のデータが到着すると新しい basic window の統計値が計算され，最も古い basic window は削除される．

相関性が高いストリームデータのペアを効率的に求めるために，StatStream では basic window 内のデータに離散フーリエ変換(DFT)を行い近似的に相関係数を求める．また，多くのペアの近似相関計算を避けるためにグリッドデータ構造を使用している．本稿ではこの sliding window を木構造で扱うことを考え，近似的に相関係数を求めるために StatStream で提案されている DFT を使った手法を用いる．

3. ストリームデータの統計値

3.1 平均，分散，相関係数

サイズ w の sliding window 上のストリームデータ $(x_0,...,x_i,...,x_{w-1})$ の平均，分散，及び二つのストリームデータ x,y の相関係数は，それぞれ以下のように表される．

平均: $\bar{x} = \frac{1}{W}\sum_{i=0}^{w-1} x_i$

分散: $\sigma^2 = \sum_{i=0}^{w-1}(x_i - \bar{x})^2$

相関係数: $corr(x,y) = \dfrac{\frac{1}{w}\sum_{i=0}^{w-1} x_i y_i - \bar{x}\ \bar{y}}{\sqrt{\sum_{i=0}^{w-1}(x_i-\bar{x})^2}\sqrt{\sum_{i=0}^{w-1}(y_i-\bar{y})^2}}$

分散が大きいストリームデータを見つけることにより，ある時間内において激しく変動しているストリームデータを見つけることができる．相関係数の高い 2 つのストリームデータを見つけることにより，連動しているストリームデータを見つけることができる．

3.2 DFT を用いた相関係数の近似

slinging window の DFT 係数 (X_i) と分散 (σ_x^2) を計算することにより，相関係数を近似的に求めことが可能であることを簡単に説明する．以下ではまず DFT とその性質について説明する．次に DFT と分散を用いた相関係数の近似方法を説明する．

3.2.1 離散フーリエ変換(DFT)とその性質

時系列データ $x = \{x_0, x_1, ..., x_{w-1}\}$ が与えられたときの離散フーリエ変換である複素数列（DFT 係数列）$X = \{X_0, X_1, ..., X_{w-1}\}$ は以下のように計算できる[2]．

$$X_F = \frac{1}{\sqrt{w}}\sum_{i=0}^{w-1} x_i e^{-j2\pi Fi/w}$$

$$F = 0,1,...,w-1 \ , \quad j = \sqrt{-1}$$

DFT の性質として系列 x と系列 y のユークリッド距離を $d(x,y)$，系列 x と系列 y の DFT を X，Y と表したとき以下のような性質がある．

$$d(x,y) = d(X,Y)$$

3.2.2 相関係数の近似方法

ストリームデータ x，y の相関係数 $corr(x,y)$ は，以下のように近似して計算できる[4]．

$$corr(x,y) = 1 - \frac{1}{2}d^2(\hat{x},\hat{y})$$
$$= 1 - \frac{1}{2}d^2(\hat{X},\hat{Y})$$
$$\approx 1 - \frac{1}{2}{d_n}^2(\hat{X},\hat{Y}) \cdots\cdots(\mathrm{i})$$

ただし $\hat{x},\hat{y}$ は以下のように，元の系列 $(x = \{x_0, x_1, ..., x_{w-1}\}$，$y = \{y_0, y_1, ..., y_{w-1}\})$ を正規化した系列であり，$d_n^2(\hat{X},\hat{Y})$ は正規化した系列 $\hat{x}, \hat{y}$ のDFT係数のうち初めの n 個のDFT係数 $(X_0, X_1, ..., X_n (n < w-1))$ のユークリッド距離を表している．

$$\hat{x} = \{\hat{x}_0, \hat{x}_1, ..., \hat{x}_{w-1}\}$$

$$\hat{x}_i = \frac{x_i - \overline{x}}{\sigma_x} \quad (i = 0,1,...,w-1)$$

$$\sigma_x = \sqrt{\sum_{i=0}^{w-1}(x_i - \overline{x})^2}$$

式(i)より $d_n^2(\hat{X},\hat{Y})$ の値が小さくなれば，ストリームデータ x，y の相関係数が大きくなるため，ストリームデータ x，y の相関性が高いといえる．つまり相関性が高いストリームデータのペアを求めたい場合は，$d_n^2(\hat{X},\hat{Y})$ が小さいストリームデータのペアを探せばよい．

X_i と $\hat{X}_i$ の関係は以下のように表すことができる[4]．

$$\begin{cases} \hat{X}_0 = 0 \\ \hat{X}_i = \dfrac{X_i}{\sigma_x} \end{cases} \cdots\cdots(\mathrm{ii})$$

式(i)，(ii)を使って，sliding window の DFT 係数 (X_i) と分散 (σ_x^2) から相関係数を近似的に求めることができる．

4. sliding window の階層構造とその管理

[3]で提案されている方法をそのまま用いてストリームデータの相関を求めようとすると，誤差が無視できないくらい大きくなることが考えられる．これは，[3]ではシングルストリームの統計値を扱うことは考えているが，相関などの複数のストリームデータの統計値を扱うことを考慮していないためである．具体的には，SWAT では Detail Coefficients をすべて格納しないことによる効率化を図っているため，格納している平均値で元のデータを近似しているが，相関を求める場合にこれをそのまま適用してしまうと，一般に誤差が大きくなってしまう．そこで本稿では，平均と分散を使うことによってストリームデータの動きを近似的に表し，そのデータの傾向を使って誤差がより小さくなるようにストリームデータの相関を求めることを考える．

本稿では図 3 のような sliding window の木構造を考え，平均，分散，DFT 係数を階層ごとに更新していく方法を議論する．各ノードには平均 $(\overline{x})$，分散 (σ^2)，DFT 係数 (X_m) を格納する．以下ではこの構造に基づく平均，分散，DFT 係数の更新方法について説明する．上位レベルの DFT 係数を求める際には，ノード内のデータの傾向を使うことによって誤差が少なくなる手法を提案する．レベル i のノードの平均，分散，DFT 係数は，$b2^i$ 個のデータが到着する毎に以下のように更新すればよい．

図 3　sliding window の階層構造
Fig.3 The layered structure of sliding window

4.1 平均の更新

レベル i のノードの平均は以下のように更新すればよい．

$\overline{x}_{L_i} = \overline{x}_{S_i}$ （ノード S_i の平均をノード L_i にシフト）

$\overline{x}_{S_i} = \overline{x}_{R_i}$ （ノード R_i の平均をノード S_i にシフト）

$\overline{x}_{R_i} = \dfrac{(\overline{x}_{L_{i-1}} + \overline{x}_{R_{i-1}})}{2}$ （レベルが1つ下のノードである L_{i-1} と R_{i-1} から R_i の平均を計算）

4.2 分散の更新

分散の更新は[5]の Combination Rule を用いて更新できる．具体的には，レベル i のノードの分散は以下のように更新すればよい．

$\sigma_{L_i}^2 = \sigma_{S_i}^2$ （ノード S_i の分散をノード L_i にシフト）

$\sigma_{S_i}^2 = \sigma_{R_i}^2$ （ノード R_i の分散をノード S_i にシフト）

$$\sigma_{R_i}^2 = \sigma_{L_{i-1}}^2 + \sigma_{R_{i-1}}^2 + \frac{b2^{i-1} \times b2^{i-1}}{b2^{i-1} + b2^{i-1}}(\overline{x}_{L_{i-1}} - \overline{x}_{R_{i-1}})^2$$

つまり，レベルが 1 つ下のノードである L_{i-1} と R_{i-1} の平均と分散を使って R_i の分散を計算すればよい．

4.3 データの傾向を考慮したDFT係数の近似計算

株式市場で発生しているストリームデータや，センサーモニターで発生している温度のストリームデータの様な時系列ストリームデータを考えた際，データは連続的に変動する．つまりデータは完全にランダムな動きはしないと考えられるため，各ノード内のデータには下記のような一定の傾向があると考えられる．

① 右肩上がりのデータ
② 右肩下がりのデータ
③ 変化が少ないデータ

ノード内のデータの傾向と分散，平均を使うことによってノード内の全ての元データを格納することなく，近似的にノード内のデータの動きを表すことができる．以下ではこのノード内のデータの傾向を使って誤差が少なくなる DFT 係数の近似計算方法を提案する．DFT 係数の誤差が少なくなれば相関係数をより正確に求めることができるようになり，相関性の高いストリームデータのペアをより正確に求めることができる．

ノード内のデータ $(x_i (0 \le i \le b-1))$ の傾向を使って，上位

レベルのDFT係数を誤差が小さくなるように更新することを考える．レベル 1 の DFT 係数はレベル 0 の平均 $(\bar{x}_0)$ と分散 (σ_0^2) を使って表 2 の様にデータ値 (x_i) を近似して求める．

表 2　ノード内のデータの傾向を使ったデータの近似

Table 2　Approximation with data tendencies in nodes

ノードのデータの傾向	x_i (0≤i≤b/3-1)	x_i (b/3≤i≤2b/3-1)	x_i (2b/3≤i≤b-1)
①右肩上がり	$\bar{x}_0 - \sqrt{\frac{\sigma_0^2}{b}}$	$\bar{x}_0$	$\bar{x}_0 + \sqrt{\frac{\sigma_0^2}{b}}$
②右肩下がり	$\bar{x}_0 + \sqrt{\frac{\sigma_0^2}{b}}$	$\bar{x}_0$	$\bar{x}_0 - \sqrt{\frac{\sigma_0^2}{b}}$
③変化が少ない	$\bar{x}_0$	$\bar{x}_0$	$\bar{x}_0$

このように近似することによってノード内のデータの傾向を上位レベルのDFT係数の計算に反映することができるため誤差が少なくなる．例えば L_0 のデータの傾向が①で R_0 のデータの傾向が③のとき R_1 の DFT 係数は以下の様にして求めることができる．

$$X_{m\_R_1} \approx \frac{1}{\sqrt{w}}\left\{\sum_{i=1}^{b/3-1}\left(\bar{x}_{L_o} - \sqrt{\frac{\sigma_{Lo}^2}{b}}\right)e^{\frac{-j2\pi Fi}{2b}} + \sum_{i=b/3}^{2b/3-1}\bar{x}_{L_0}e^{\frac{-j2\pi Fi}{2b}} + \sum_{i=2b/3}^{b-1}\left(\bar{x}_{L_o} + \sqrt{\frac{\sigma_{Lo}^2}{b}}\right)e^{\frac{-j2\pi Fi}{2b}} + \sum_{i=b}^{2b-1}\bar{x}_{R_0}e^{\frac{-j2\pi Fi}{2b}}\right\}$$

レベル2のノードのDFT係数を計算する方法は以下の2種類の方法が考えられる．

- レベル1の平均と分散だけを使って計算する
- レベル1とレベル0の平均と分散を使って計算する

どちらの方法で計算したほうが誤差がより小さくなるかは，L_0 と R_0 のノード内のデータの傾向による．L_0 と R_0 のデータが①＋②，②＋①の様に上下または下上という傾向を示しているときは，レベル1 (L_1) とレベル0 $(L_0$ と $R_0)$ の平均と分散を使ってレベル2のDFT係数を求めたほうが誤差は少なくなる．これは L_0 と R_0 のデータが上下，下上という傾向を示しているにもかかわらず，レベル1の平均と分散だけを使ってデータを近似するとほとんど変化がないデータと見なされてしまうからである．また①＋①，②＋②，③＋③の様に同じ傾向を示しているときは，レベル1の平均と分散だけを使ってレベル2のDFTを求めたほうが誤差は少なくなる．その他の場合（①＋③，②＋③，③＋①，③＋②）は，レベル1の平均と分散だけを使ってレベル2のDFT係数を求める．

同じようにしてレベル i のDFT係数を求めるとき，レベル $i-2$ のノード L_{i-2} と R_{i-2} のデータの傾向が上下，下上のときはレベル $i-1$ (L_{i-1}) とレベル $i-2$ $(L_{i-2}$ と $R_{i-2})$ の平均と分散を使って求める．その他の場合はレベル $i-1$ $(L_{i-1}$ と $R_{i-1})$ の平均と分散を使って求める．

以上のようにノード内のデータの傾向を使うことにより，誤差を少なくしてDFT係数を効率よく計算することが可能となる．

4.4 DFT係数の更新

レベル i のノードのDFT係数は以下のように更新される．

$$X_{m\_L_i} = X_{m\_S_i}$$

$$X_{m\_S_i} = X_{m\_R_i}$$

4.3で示したようにレベル i より下のノードの平均, 分散を使って新しい $X_{m\_R_i}$ を計算する．

4.5 考察

図3のようなsliding windowの木構造を考えることによって以下のような利点がある．[2]の方法を使ってsliding window内のDFT係数を求める際， k をsliding windowの分割数とすると， O(k) 個のデータを参照しなくてはならない．一方本稿の方法ではO(log k) 個のデータを参照すれば良いため，[2]よりもDFT係数の更新が速くなると考えられる．精度についてはノード内のデータ傾向を考慮しているため，[3]をそのまま使ってDFT係数を求めるより良くなると考えられる．ただし本稿の方法は最新のデータの誤差は少なくできるが，過去のデータについては[2]より誤差が大きくなることが考えられる．このため[2]の方法と比べた場合，精度がどのようになるかは今後の課題である．

5. まとめと今後の課題

本稿ではストリームデータの相関を効率良く求めるために，階層的な DFT 係数の計算方法を提案した．この DFT 係数の計算方法は，ノード内のデータの傾向を使うことにより誤差を少なくすることが可能となっている．今後は，提案手法をシステムとして実装し，具体的な応用分野に適応してその有効性を検証していく予定である．

[文献]

[1]B.Babcock, S.Babu, M.Datar, R.Motwani and J.Widom.: “Models and issues in data stream systems”, Proc. PODS, pp.1-16 (2002).

[2]Y.Zhu and D.Shasha: “StatStream: Statistical monitoring of thousands of data streams in real time”, Proc. VLDB, pp.358-369 (2002).

[3]A.Bulut and A.K.Singh: “SWAT: Hierarchical stream summarization in large networks”, Proc. ICDE, pp303-314 (2003).

[4]R.Agrawal, C.Faloutsos and A.N.Swami: “Efficient similarity search in sequence databases”, Proc. FODO, pp.69-84 (1993).

[5]B.Babcock, M.Datar, R.Motwani and L.O'Callaghan: “Maintaining variance and k-medians over data stream windows”, Proc. PODS, pp234-243 (2003).

畑中　洋介　Yousuke HATANAKA

群馬大学大学院工学研究科博士前期課程在学中．2002 群馬大学工学部情報工学科卒業．ストリームデータ処理に興味を持つ．

有次　正義　Masayoshi ARITSUGI

群馬大学工学部情報工学科助教授．1991 九州大学工学部情報工学科卒．1996 同大大学院博士後期課程了．博士(工学)．データベースシステム，分散並列データ処理等に興味を持つ．