

教師なし学習における 非データ分布依存型コンセプトドリフト検出手法の検証

坂本 悠輔¹ 福井 健一² Joao Gama³ Daniela Nicklas⁴ 森山 甲一² 沼尾 正行²

概要:近年、コンピュータやセンサ技術の発達からデータをデータストリームとして連続的に獲得できる環境が増加している。しかし、現在これらの豊富なデータから十分に知識を獲得できているとは言えない。データストリームに対して学習手法を用いて知識の獲得を試みる際には、時間の経過と共に学習すべき概念が変化すること、つまりコンセプトドリフトに対応する必要がある。そのための方法の一つとしてコンセプトドリフト検出手法がある。これまで、コンセプトドリフト検出手法の研究は教師あり学習を中心に行われてきた。教師なし学習においても研究は行われ、一定の成果は取めたが計算量が大きいという欠点があった。そこで、本研究では、教師なし学習において計算量の少ないコンセプトドリフト検出手法を目指して、教師あり学習と信号処理において効果が確認されているコンセプトドリフト検出手法を応用し実験を行った。その結果、今後のコンセプトドリフト検出手法の開発において重要な知見を得た。

1. はじめに

近年、日常的に利用する様々なモノがインターネットに接続され“モノのインターネット”が急速に増加している。このような背景から、データストリーム処理に対する関心が高まっている。データストリームから知識を獲得するためには、時間の経過と共にデータの特徴が変化することに対応することが必要不可欠である [1][2]。

コンセプトドリフトへの対応としては二種類考えられる。一つ目の方法は、コンセプトドリフトが起こったかどうかに関係なく、随時モデルを更新するオンライン学習である。しかし、この方法では、モデル更新が頻繁に行われるため、学習モデルへのラベリングなどの人の介入するプロセスを組み込むことが困難である [3]。二つ目の方法は、コンセプトドリフトを検出し、その後モデルを更新する方法である [4][5][6]。この方法では、学習モデルの更新が必要最小限におさえられるため、人の介入するプロセスを組み込むことが可能になる。本研究では、学習モデルへのラベリングを含むモニタリングシステムの開発を見据えているため、

二つ目の方法に着目している。これまで、コンセプトドリフト検出に関する研究は、Gama らを中心に主に教師あり学習の下で進められてきた [5]。しかし、データへの明確なラベル付与が困難な場面を想定すると、教師なし学習の下でのコンセプトドリフト検出手法の確立は重要である。データの分布全体を計算する手法が提案されているが、計算コストが大きい [4][6]。例えば、モノの異常や危険に対するモニタリングシステムを考えた場合、計算コストの大きさは異常の検知の速度として反映される。よって、教師なし学習においても非データ分布依存型のコンセプトドリフト検出手法を確立することは重要である。そこで本研究では、教師あり学習の下で有効性が確認されている非データ分布依存型の統計量に基づくコンセプトドリフト検出手法を教師なし学習の下で応用し、その有効性を検証した。本研究では、コンセプトドリフト検出手法を用いた固体酸化物型燃料電池 (SOFC) 内に発生する損傷に対するモニタリングシステムを開発することを見据えている。ここでは、損傷が発生する際に同時に放出される超音波データを用いてモニタリングすることを想定しており、コンセプトドリフト検出手法は超音波データの変化（損傷の変化）を発見するために用いられる。

2. コンセプトドリフト

はじめにコンセプトドリフトとは、データストリームを

連絡先: 福井 健一, 大阪大学産業科学研究所, 〒 567-0047 大阪府茨木市美穂ヶ丘 8-1, TEL:06-6879-8426, FAX:06-6879-8428, mail:fukui@ai.sanken.osaka-u.ac.jp

¹ 大阪大学大学院情報科学研究科
Graduate School of Information Science and Technology, Osaka University

² 大阪大学産業科学研究所
The Institute of Scientific and Industrial Research, Osaka University

³ University of Porto, Portugal

⁴ University of Bamberg, Germany

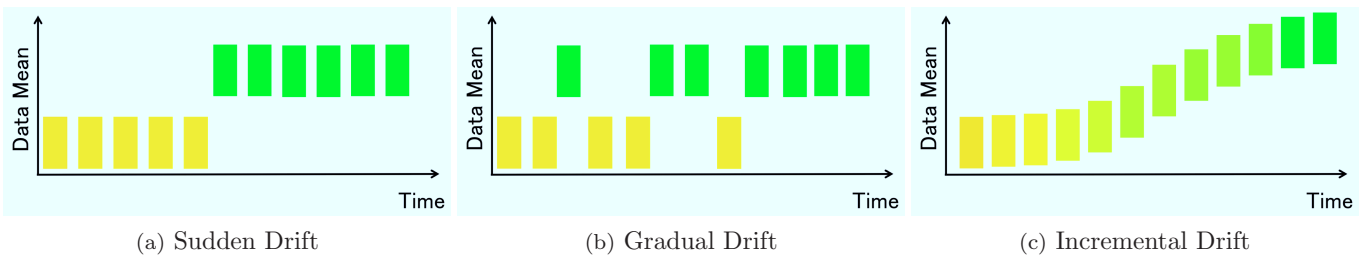


図1 コンセプトドリフトの種類

扱う際に起こる問題である。データストリームとは、センサなどのアプリケーションから継続的にかつ時系列順に得られるデータのことである [7]。コンピュータの発達やインターネットの普及に伴い、現在、データストリームの中に潜む情報を発掘することの重要性は増している。多くの場合、このデータストリームは現象の変化、劣化、故障、事故などにより時間が経過すると変化する。この現象をデータストリームの中から情報を発掘するアプリケーションの側から考えると、データストリームの変化に伴い、学習すべき概念（コンセプト）が変化するということに相当する。このような時間経過に伴う学習すべき概念（コンセプト）の変化がコンセプトドリフトである [1][2]。より正確に述べれば、情報源 S の観測量 X の確率分布 $Pr(X|S)$ の変化や新たな情報源 S' の確率分布 $Pr(X|S')$ の出現を表す。

2.1 コンセプトドリフトの種類

コンセプトの変化の仕方にはいくつか種類が存在する。アプリケーションにとって、コンセプトの様々な変化の仕方に対応することは非常に重要である。そこで、ここではコンセプトドリフトの種類について SOFC の損傷 [8] に対するモニタリングを例にとって述べる。

• Sudden Drift

Sudden Drift は最も単純な変化であり、ある時刻 t にデータストリームの情報源が S_1 から S_2 へと変化するることによって観測量の確率分布が突然 $Pr(X|S_1)$ から $Pr(X|S_2)$ へ変換することである (1次元データの概念図を図1(a)に示す)。Sudden Drift の例としては、SOFC のモニタリングにおいて考えると SOFC 内のセルを交換した場合や突発的な故障が挙げられる。

• Gradual Drift

Gradual Drift は突然の変化ではなく、ある程度時間をかけてデータストリームの情報源が S_1 から S_2 へと徐々に変換することである (図1(b))。時間の経過と共に徐々に $Pr(S_1)$ が減り、代わりに $Pr(S_2)$ が増え最終的にすべてのデータは $Pr(S_2)$ から取得されるようになるような変化である。SOFC のモニタリングにおいて考えると、これは損傷の進展が考えられる。SOFC 内の損傷は進展することが

知られており、例えば、電解質の亀裂からガラスシールの損傷のように損傷が進展する。この損傷の進展は一般にゆるやかなことが過去の研究から知られている [8]。

• Incremental Drift

Incremental Drift は情報源の変化ではなく観測量の確率分布 $Pr(X|S)$ がわずかな変化を繰り返し最終的に全く異なるデータ分布へと変化するコンセプトドリフトである (図1(c))。そのため、Incremental Drift を検出するまでに長い時間を要する場合が多い。Incremental Drift は、SOFC のモニタリングで考えると温度や酸化の進展による変化などにより材質が変化し、それが超音波データに反映され徐々にデータが変化する場合が挙げられる [8]。

2.2 コンセプトドリフトへの対応

コンセプトドリフトへの対応方法は、学習モデルの更新方法の違いによって次の2つの方法に分けられる [1][2]。

- (1) **Trigger method:** 学習モデルの更新はデータ分布の変化を検出して始めて行われる。
- (2) **Evolving method:** 学習モデルの更新は、コンセプトドリフトの有無にかかわらず定期的に行われる。

Trigger method はモデル更新をする必要があるというサインを受け取って始めてモデルが更新される。それに対し、Evolving method ではコンセプトドリフトが起こったかどうかは問題とせず、モデル更新を頻繁に行うことによってデータ分布に変化が起こったとしても対応しようとする方法である。

本研究では、Trigger method によってコンセプトドリフトに対応することを目指す。その最大の理由は、本研究では SOFC の損傷に対するモニタリングシステムの開発を見据えていることにある。このモニタリングシステムでは、自己組織化マップ (Self-Organizing-Map, SOM)[9] をクラスタリング手法として用い、ラベル付けのプロセスを含む。そのため、モデル更新があまりにも頻繁に行われてしまうと人が介入できなくなってしまう。一方で、人が常に介入できる頻度でモデル更新をすることはコンセプトドリフトに追従できない危険性をはらむ。よって、本研究では、モデル更新を必要最小限に抑えることのできるコンセプトドリフト検出手法 (Trigger method) が適すると考えられる。

2.3 教師なし学習におけるコンセプトドリフト検出手法

コンセプトドリフト検出手法は、データ分布そのものを求めるか否かという点で大別され、これまで教師なし学習においては、データ分布を計算し、分布の変化を検出する方法が提案されている [4][6]。この方法の優れた点は、データの分布そのものを知ることができるため様々なコンセプトドリフトに対応可能であることである。しかし、一方で計算量が大きいという欠点も併せ持つ。モニタリングシステムにおいて、計算量は異常検知の遅れとして反映されるため小さい計算量が望ましいが、これまでそのような方法は提案されていない。

そこで、本研究では、非データ分布依存型の統計量に基づくコンセプトドリフト検出手法に着目し検証を行った。

2.4 本実験で検証する非データ分布依存型コンセプトドリフト検出手法

本研究では、教師あり学習や信号処理におけるコンセプトドリフトに対し有効性が確認されている手法を教師なし学習において応用することを目指した。ここでは、それらを紹介する。

・ Static Process Control 法 (SPC 法)

SPC 法は Gama らによって提案された教師あり学習の二値分類に対するコンセプトドリフト検出手法である [5]。この手法の特徴は、データ分布全体の分布を計算することなくコンセプトドリフトを検出している点である。以下に、SPC 法におけるコンセプトドリフト検出の方法を述べる。まず初めに、入力データ (\vec{x}_i, y_i) が連続して得られる状況を考える。ここで、 \vec{x}_i は特徴ベクトル、 y_i はそれに対応するクラスラベル、 i はデータ番号である。さらに、決定モデル（決定木やナイーブベイズ分類器など）がそれぞれの入力データに対し \hat{y} を予測する。ここで、 $y_i = \hat{y}$ である場合が正解であり、 $y_i \neq \hat{y}$ である場合、誤りである。つまり、各入力データにおいてエラーはベルヌーイ試行によって決定される。よって、ここで各入力データに対し false probability p_i を考えると、 p_i は二項分布に従うため p_i の分散 s_i は次の様に計算される。

$$s_i = \sqrt{p_i(1-p_i)/i} \quad (1)$$

SPC 法では、この p_i と s_i の 2 つの値が p_{min} , s_{min} として保存されコンセプトドリフト検出に用いられる。これら二つの値は、入力データが与えられ、 $p_i + s_i$ が $p_{min} + s_{min}$ を満たす時に更新される。また、標本数が十分である場合、二項分布は標準正規分布に近似して考えることができる。ここで、コンセプトドリフトが起きない限りデータ分布が変化することはないと考えると信頼区間 $1 - \delta/2$ は $p_i \pm \alpha * s_i$ と近似できる。ここで、 α は信頼係数である。

例えば、 $\alpha = 2$ とした場合、標準正規分布の信頼区間およそ 95% に対応し、 $\alpha = 3$ とした場合、およそ 99% に対応する。このことを利用して、SPC 法ではコンセプトドリフト検出に向けて 2 つの閾値が準備される。1 つ目の閾値は、Warning level である。Warning level は信頼区間 95% の近似を利用して $p_i + s_i \geq p_{min} + 2 * s_{min}$ と設定される。2 つ目の閾値は Drift level である。これは、信頼区間 99% の近似を利用して $p_i + s_i \geq p_{min} + 3 * s_{min}$ と設定される。ここで、ある入力データ x_w が Warning level に達したとすると、それ以降の入力データはコンセプトドリフトが起こった場合に備えて新しい学習モデル作成のために保存される。そして、その後のある入力データ x_d が Drift level に達しとすると、この時コンセプトドリフトが検知される。そして、 x_w から x_d までの保存されたすべてのデータを使って、新しい学習モデルが作成される。また、この時 x_w と x_d はリセットされる。Algorithm 1 に、SPC 法のアルゴリズムを示す。

・ 分散計算の変更

本研究で SPC 法を用いるに当たって、各入力データの分散計算に変更を加えた。先に述べたように SPC は、本来教師あり学習の二値分類に対するコンセプトドリフト検出手法である。そのため、入力データの false probability p_i の標準偏差 s_i は二項分布に従い、(1) 式に従って計算される。しかし、本研究では SOM を用い、 p_i は入力データの SOM 学習モデルへの割り当て誤差とした。そのため、 p_i は二項分布に従わず (1) 式を用いることはできない。SOM における入力データ x_i の割り当て誤差 p_i は次のように計算される。

$$p_i = \min_j \|x_i - W_j\| \quad (j = 1, 2, \dots, m) \quad (2)$$

ここで、 W_j は SOM の第 j ニューロン（ノード）の参照ベクトルを表し、 m はノード数である。そして、 p_i に対し $p_{i-w}, p_{i-w+1}, \dots, p_i$ (w はウィンドウ幅) のようなウィンドウを用いることによって標準偏差 s_i を次のように計算した。

$$s_i = \sqrt{\frac{\sum_{n=i-w}^i (p_n - \mu)^2}{w+1}} \quad (3)$$

ここで、 μ はウィンドウ内の p_i の平均である。

・ Page-Hinkley test 法 (PHT 法)

PHT 法はこれまで変化点の検出に多く用いられてきた逐次解析手法である [10][11][12]。とくに、確率変数（電圧値など）が正規分布に従うガウス信号の平均の変化を検出することを得意としている。この手法では、観測開始からの観測値の累積誤差 U_T が次のように計算される。

$$U_T = \sum_{t=1}^T (x_t - \bar{x}_T - \sigma) \quad (4)$$

ここで、 x_t は観測値、 \bar{x}_T は、 $\bar{x}_T = 1/T \sum_{t=1}^T x_t$ と決定さ

Algorithm 1 Statical Process Control 法 (SPC 法)

Input:

labeled dataset x_1, x_2, \dots, x_t
warning threshold t_w (default $t_w = 2$)
detection threshold t_d (default $t_d = 3$)
warm-up window size w_0 (default $w_0 = 30$).

1. Initialize the minimum classification error $p_{min} = \infty$ and the corresponding standard deviation $s_{min} = \infty$. Set the warning zone flag, f_w , to false and $w_1 = 0$.

2. For $j = 1$ to $t - 1$ (all the observations)

if $j < w_0$ **then then**

$w_{j+1} = w_j + 1$ (warm up, only grow the window)

else

i. Train a classifier on the current window of size w_j .

ii. Classify observation w_{j+1} .

iii. Update the error rate over the current window.

Let \hat{p} be the updated error rate and $\hat{s} = \sqrt{\hat{p}_i(1 - \hat{p}_i)/w_j}$ be the updated standard deviation.

iv. If $(\hat{p} + \hat{s}) < p_{min} + s_{min}$ then update the minimum error by $p_{min} = \hat{p}$ and $s_{min} = \hat{s}$

v. If $(\hat{p} + \hat{s}) \geq (p_{min} + t_d * s_{min})$ and $f_w = true$ Change has been detected, set up the detection time $t_d = j$

Take as the new training and detection window all the observations

since t_w (size $w_{j+1} = j - t_w + 1$), set $p_{min} = \infty$, $s_{min} = \infty$ and $t_w = \infty$.

ElseIf $(\hat{p} + \hat{s}) \geq (p_{min} + t_w * s_{min})$

If $f_w = false$

switch the warning zone flag $f_w = true$ and set up the warning time $t_w = j$.

Else

set $f_w = false$ and update the window by adding x_{j+1} to it (size $w_{j+1} = w_j + 1$)

end if

$DT_{SPC} = t_w$

Output: detection time DT_{SPC}

れる観測開始からの観測値の平均である。また、 σ はどの程度の変化をノイズとして許容するかを決定するパラメータである。また、 m_T が、 U_T の最小値として保存される ($m_T = \min(U_1, U_2, \dots, U_T)$)。そして、 U_T と m_T を用いて PH_T が $PH_T = U_T - m_T$ と決定される。 PH_T の値が事前に設定されたパラメータ λ よりも大きくなった場合、コンセプトドリフトと判断される。つまり、PHT 法においてコンセプトドリフトと判断されるのは次の式を満たす時である。

$$PH_T > \lambda \quad (5)$$

Algorithm 2 に、PHT 法のアルゴリズムを示す。

3. 実験

本研究では、上に挙げた非データ分布依存型の SPC 法、

Algorithm 2 Page-Hinkley test 法 (PHT 法)

Input:

labeled dataset x_1, x_2, \dots, x_t
magnitude threshold σ
detection threshold λ

for $t > 0$ **do**

1 Computes

$$\bar{x}_T = 1/T \sum_{t=1}^T x_t$$

$$U_T = \sum_{t=1}^T (x_t - \bar{x}_T - \sigma)$$

$$m_T = \min(U_1, U_2, \dots, U_T)$$

If $PH_T = U_T - m_T > \lambda$

return and report a change at time t_{PH}

else

return to 1

end for

Output: detection time t_{PH}

PHT 法の 2 手法が教師なし学習におけるコンセプトドリフト検出法として有効であるかを検証することを目的として人工データを用いて実験を行った。

3.1 人工データ

本研究では、2 次元の人工データを用いて実験を行った。なお、データは 2 つの中心の異なる標準正規分布に従って生成された。生成された人工データの例を図 2 に示す。図 2 は正規分布の中心間のクラス間距離 1.5 のものである。実験ではクラス間距離を変化させた。

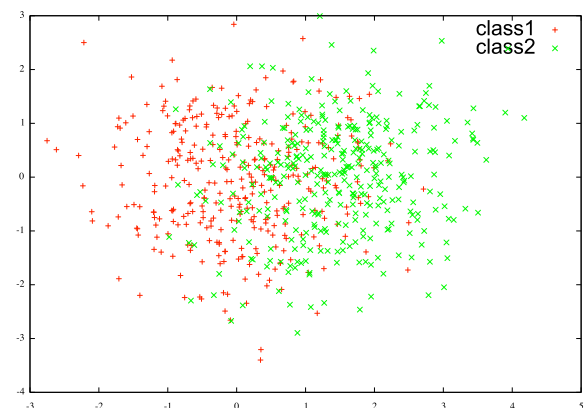


図 2 人工データの生成

3.2 実験手順

実験は以下のような手順で行った。

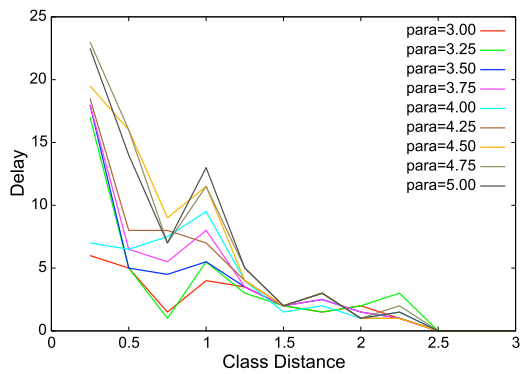


図 3 SPC 法におけるクラス間距離と検出遅れの関係

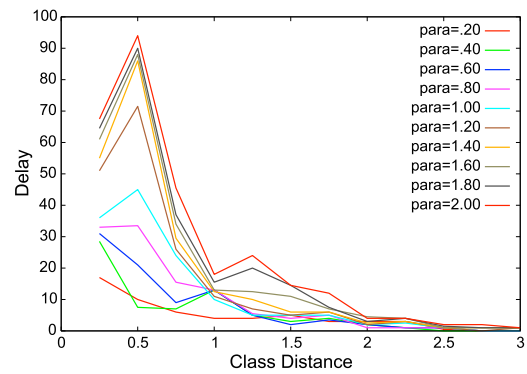


図 4 PHT 法におけるクラス間距離と検出遅れの関係

実験手順

- (1) クラス 1(350 点) とクラス 2(350 点) を中心の異なる標準正規分布に従って生成する。
- (2) 生成されたクラス 1(300 点) を用いて SOM 学習モデルを作成する。
- (3) クラス 1(50 点) → クラス 2(350 点) の順でテストデータとして学習モデルに割り当て、各手法によってコンセプトドリフトの有無を確認する。
- (4) 1~3 を複数回試行する。
- (5) クラス 1,2 のクラス間距離を変更し、1~5 を行う。

3.3 実験 1. 検出遅れとクラス間距離の関係

まず、初めにクラス間距離を変化させた時に各コンセプトドリフト検出手法においてどのような変化が起きるかを検証する実験を行った。クラス間距離を 0.25, 0.50, 0.75, ..., 3.00 と変化させた。また、各距離における試行回数は 30 回とした。また、テストデータセット中のクラス 1 のデータ 50 点としたため、コンセプトドリフトはデータ割り当て番号 50 において発生し、最速のコンセプトドリフト検出はデータ割り当て番号 51 であり、この時を検出遅れ 0 とした。以下に、各手法における設定とそれぞれの結果を記す。

・SPC 法

ウィンドウサイズは 10 とし、パラメータ α は、3.00, 2.25, 2.50, ... , 5.00 と変化させた。

結果を図 3 に示す。横軸がクラス間距離を示し、縦軸は各クラス間距離における検出遅れを示す。なお、ここで検出遅れは 30 回の試行の中央値である。PHT 法においても同様にして結果を示す。

・PHT 法

今回の実験では、PHT で用いられる 2 つのパラメータのうち σ は変化させず、常に $\sigma = 0.05$ とした。この値は先行研究でよく用いられているものである。一方、 λ は、0.2, 0.4, 0.6, ... , 2.0 と変化させた。結果を図 4 に示す。

図 3, 図 4 より、2 手法どちらにおいてクラス間距離が十分に大きい時 (つまりコンセプトドリフト検出の難易度が低い時)、検出遅れが小さくなることを確認した。この

ことから、コンセプトドリフトが顕著に現れた時点では、教師なし学習においても有効なコンセプトドリフト検出手法として機能することを確認した。

また、図 3, 図 4 のすべてでクラス間距離 1.0 付近を境に検出遅れが減少していることが見てとれる。今回の実験で用いた人工データは標準正規分布に従って生成されているため、クラス間距離 1 は 1σ にあたる。 1σ より難易度が高い範囲では、パラメータによっては検出遅れが小さくなっているが、次の実験で示すように誤検出数とのトレードオフが存在する。

3.4 実験 2. ウィンドウサイズとコンセプトドリフト検出性能の関係

実験 1 では、SPC 法のウィンドウをウィンドウサイズ 10 に固定して実験を行った。そこで、今回の実験では、ウィンドウサイズによるコンセプトドリフト検出性能への影響を検証するために SPC 法で用いるウィンドウサイズを変化させて実験を行った。

ウィンドウサイズは 10, 30, 40 と変化させた。また、PHT 法に対しても引き続き実験を行った。この時、PHT 法のパラメータに変更は加えなかった。また、前回の実験より、データ分布の変化が 1σ 付近の変化を調べれば十分であるということが確認できたため、クラス間距離は 0.5, 1.0, 1.5 と変化させた。また、各距離における試行回数を 50 回とした。各距離におけるパレート解析の結果を図 5, 図 6, 図 7 に示す。

すべての距離において、適切なサイズのウィンドウを用いた時 SPC 法は PHT 法よりも良いコンセプトドリフト検出性能を示すことが確認できる。SPC 法のウィンドウサイズを小さくするとパレート解の数が少なくなっていることが確認できる。これは、後検出と検出遅れにおいて多様な解が得られないことを示している。

また、クラス間距離が短い時、大きいサイズのウィンドウを用いたほうが良い検出性能を示した。一方、クラス間距離が大きい時はパラメータ次第でウィンドウサイズには依存しないことが確認できる。まず誤検出数は、ウィンド

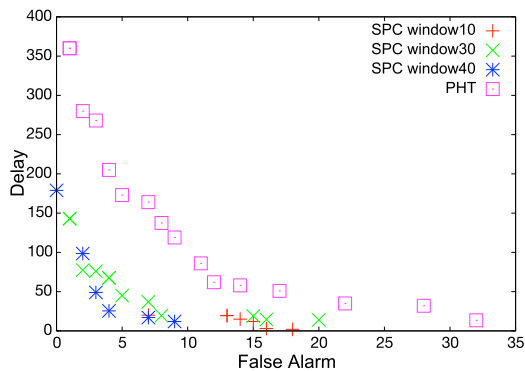


図 5 距離 0.5 における各手法の誤検出数と検出遅れの関係

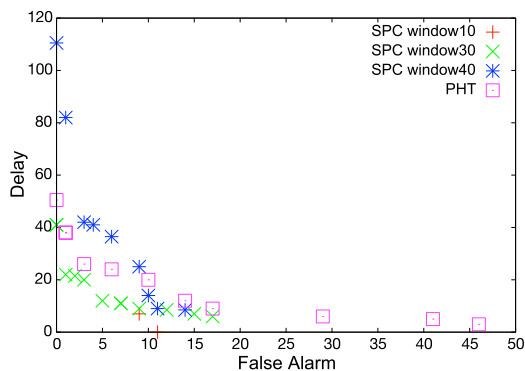


図 6 距離 1.0 における各手法の誤検出数と検出遅れの関係

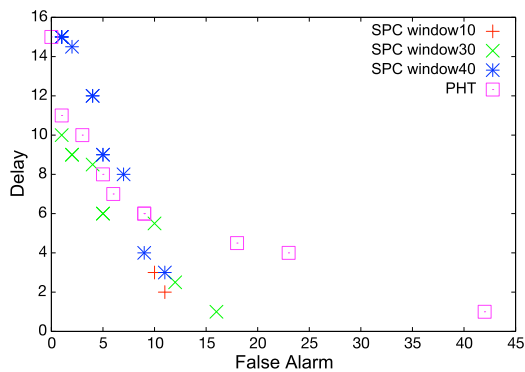


図 7 距離 1.5 における各手法の誤検出数と検出遅れの関係

ウサイズとパラメータのみに依存し、特に大きいウィンドウを用いるほど誤検出数を抑えることができる。しかし、同時に検出遅れは増加してしまう。クラス間距離が小さい場合には大きいウィンドウサイズを用いたほうが検出遅れを抑えることができ、誤検出数と合わせて考えた時最も効果的な検出方法となる。よって、クラス間距離が小さい時には、大きいウィンドウを用いてパラメータを適切に設定することが効果的である。

4. まとめ

本研究では、教師なし学習の下でのコンセプトドリフト検出において、非データ分布依存型の統計量に基づく方法を検討した。今回行った2つの実験の結果から、適切なウィンドウサイズを設定すればSPC法のほうがPH法よ

り誤検出および検出遅れが小さいことがわかった。また、異なるガウス分布のクラス間距離 1σ になると、検出閾値のパラメータやウィンドウ幅の影響が大きく現れることを確認した。さらに、データ分布の変化が小さいコンセプトドリフトほどSPC法において用いるウィンドウサイズを大きくすることが有効であることが確認できた。

参考文献

- [1] Indre Zliobaite: Learning under Concept Drift: an Overview, <http://arxiv.org/abs/1010.4784> (2010)
- [2] João Gama and Indre Žliobaitė and Albert Bifet and Mykola Pechenizkiy and Abdelhamid Bouchachia: A Survey on Concept Drift Adaptation, *ACM Computing Surveys*, Vol. 46, No. 4 (2014)
- [3] Jonathan A. Silva and Elaine R. Faria and Rodrigo C. Barros and Eduardo R. Hruschka and Andre C. P. L. F. De Carvalho and Joao Gama: Data Stream Clustering: A Survey, *ACM Computing Surveys(CSUR)*, Vol. 46, No. 1 (2013)
- [4] Rosane M.M. Vallim, José A. Andrade Filho, Rodrigo F. de Mello, André C. P. L. F. de Carvalho1, João Gama: Unsupervised density-based behavior change detection in data streams, *Intelligent Data Analysis*, Vol.18, No.2, pp.181-201 (2014)
- [5] João Gama, Pedro Medas, Gladys Castillo, Pedro Rodrigues: Learning with Drift Detection, *Proceedings of 17th Brazilian Symposium on Artificial Intelligence*, pp. 286-295 (2004)
- [6] Guénaél Cabanes and Younès Bennani: Change detection in data streams through unsupervised learning, *WCCI 2012 IEEE World Congress on Computational Intelligence*, pp.1- 6 (2012)
- [7] Hans-Jürgen Appelrath, Dennis Geesen, Marco Grawunder, Timo Michelsen, and Daniela Nicklas: Data Stream Clustering: Odysseus: a highly customizable framework for creating efficient event stream management systems, *In Proceedings of the 6th ACM International Conference on Distributed Event-Based Systems*, pp. 367-368 (2012)
- [8] 福井健一, 赤崎省悟, 佐藤一永, 水崎純一郎, 森山甲一, 栗原聡, 沼尾正行: 固体酸化燃料電池における損傷過程可視化, *日本機械学会論文集 A 編*, Vol. 76, No. 762, pp. 223-232 (2010)
- [9] T. Kohonen: “Self-Organizing Maps”, Springer-Verlag (1995)
- [10] Hinkley D.: Inference about the change point in a sequence of random variables, *Biometrika*, 57(1), 1-17(1969)
- [11] Mouss, H., D. Mouss, N. Mouss, and L. Sefouhi :Test of Page-Hinkley, an approach for fault detection in an agro-alimentary production system. *In Proceedings of the Asian Control Conference*, Volume 2, pp.815-818 (2004)
- [12] Hartl, C., N. Baskiotis, S. Gelly, and M. Sebag: Change point detection and metabandits for online learning in dynamic environments. *In Confrence Francophone sur l' apprentissage automatique*, pp.237-250, Cepadues (2007)