307

オンラインサポートベクトル回帰におけるリアルタイム予測の検討

室蘭工業大学 ○ 杉本 大志，倉重 健太郎

要 旨

非線型サポートベクトル回帰では，過去のカーネル行列から統計的な予測が可能だが，学習データを逐次追加した場合，再学習すると同時にすべてのデータを保存する必要がある．この問題の解決策のひとつとして，逐次学習によって再学習を回避した Online SVR が提案されているが，逐次的な予測の手法が示されておらず，リアルタイム制御への応用が難しい．

本稿では，この問題を解決することで，Online SVR 上でリアルタイムに逐次予測する手法を提案し，その有用性を検討する．

1. はじめに

機械学習の分野はここ数年で拡大しており，この成果を利用することで数多くの新技術も成長を遂げている．機械学習における既存のアルゴリズムのうち最も有名であり，最近のトレンドとなっている理論として，識別問題のためのサポートベクトルマシン (SVM)[1]，回帰分析のためのサポートベクトル回帰 (SVR)[2] があげられる．本来の SVM, SVR は線型分離問題，線型回帰分析のみを対象に考案されたものであるが，Kernel Trick[3] と呼ばれる手法を適用することで応用範囲はさらに拡大し，注目されるようになった．

最近は，識別問題への適用にとどまらず，ロボットの制御，認識に SVM や SVR(以下，一括学習による SVR とよぶ) を適用した研究が行われている．小林らは，一括学習による SVR を用いて，倒立振子の逆システムを学習させた [4]．この研究では，あらかじめ状態フィードバック制御で得られた入出力データを用いて学習し，これにより得た SVR モデルを姿勢制御に適用することで，SVR が初期値に対してロバストであり，ロボットの学習制御への応用に有効であることを，シミュレーションによって示した．

この SVM, SVR の欠点の一つとして，一括学習でのみ動作するということがあげられる．これらの手法では，すべての学習データを用いて学習し，新たな学習データが追加された場合は，すでに存在する学習データに加えて再学習することが必要となる．この問題については様々な解決策が提案されている．そのなかでも Parrella は，逐次学習の手法を SVR に適用した Online SVR[5] を提案している．このアルゴリズムを適用することにより，学習器が最初から再学習をすることなく学習データの追加 (追加学習) と削除 (忘却) を行うことができるとしている．

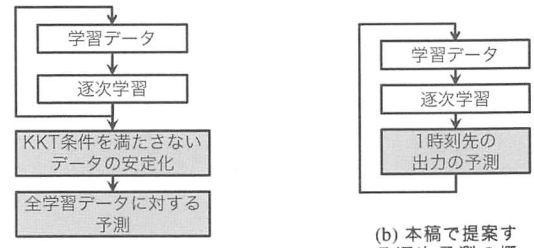
しかし Parrella は，自身の手法によると，回帰モデル $y = \mathbf{w}^\top \mathbf{x} + b$ に含まれる重み係数ベクトル $\mathbf{w}$ を計算する際，浮動小数点演算を行うことで数値的不安定性が生じる可能性があるとしている．これによる演算誤差により，計算結果に不適切な大きな数値を出力することがあると指摘している．この問題については，KKT 条件 [6] を満たさなかった学習データを忘却し，これまでに追加された学習データ系列の最後尾にその学習データ追加し改めて学習させるということで対処している [5](図 1(a)，またこの操作を “Stabilization[5]”，安定化と称している)．このため，安定化を行うことなく逐次学習を行った後予測した場合，各学習データの値から逸脱した結果を示す可能性が考えられる．また，この手法では，再学習を避ける形で学習データを逐次的に与えることは可能だが，次時刻における出力の予測を，逐次的に与える方法については述べられていない．このため，ロボット制御などに代表されるリアルタイム制御へ応用するには，あらかじめ入出力データを与えモデルを学習させるのではなく，学習データが時間とともに変化することを考慮し，データを逐次与えて学習させると同時に，次時刻の出力を予測する手法を与える必要がある．

そこで本稿では，Parrella による $\mathbf{w}$ の計算手法を利用しないことで，数値的不安定性を回避することを考える．そして，逐次的に学習データを与え Online SVR により学習させると同時に，これまでの学習から得た過去のサポートベクトルと，新たに入力された学習データ，あわせて学習により更新されるパラメータを用いることで，予測式を構成する．この予測式を適用し，逐次学習と組み合わせることにより，Parrella の提案では述べられていなかった逐次予測の手法を提案する．

2. 逐次予測の構成

2.1 提案手法の流れ

本稿では，Online SVR が学習データを逐次取得し学習すると同時に，学習における更新式をもとに次時刻における出力 (以下，次状態とよぶ) を予測する手法を提案する．本手法の流れを図 1(b) に示す．両者を比較すると，本稿で述べる提案手法では学習結果に対する安定化を行わず，また逐次学習とともに次状態の予測を実施している．



(a) **Parrella** が提案した手法の概要

(b) 本稿で提案する逐次予測の概要

図 1 システム構成の比較

2.2 新入力に対する逐次予測の考え方

新入力 $\mathbf{x}_{N+1}$ に対する予測値 $\hat{y}_{N+1}$ を，センサなどより得た過去の観測値 $\mathbf{y} = [y_1, \ldots, y_N]^\top$ からなるデータを用い推定する．一括学習による SVR において $\mathbf{x}_{N+1}$ に対する $\hat{y}_{N+1}$ は，

$$\begin{aligned}\hat{y}_{N+1} &= \mathbf{w}^\top \phi(\mathbf{x}_{N+1}) + b \qquad (2.1)\\ &= \mathbf{a}^\top \mathbf{\Phi} \phi(\mathbf{x}_{N+1}) + b\\ &= \mathbf{k}(\mathbf{x}_{N+1})^\top (\mathbf{K} + \lambda \mathbf{I}_N)^{-1} \mathbf{y} + b \qquad (2.2)\end{aligned}$$

で予測できることが知られている．ここで，b はバイアス項，λ は正則化パラメータ，$\mathbf{\Phi}$ はデザイン行列と呼ばれ，その第 n 列は $\phi(\mathbf{x}_n)^\top$ であり，

$$\mathbf{w} = \sum_{i=1}^{N} (a_i - a_i^*)\, \phi(\mathbf{x}_i) = \sum_{i=1}^{N} \theta_i \phi(\mathbf{x}_i) = \mathbf{\Phi}^\top \mathbf{a} \qquad (2.3)$$

$$b = \epsilon + y_i - \sum_{i=1}^{N} \theta_i k(\mathbf{x}_i^\top \mathbf{x}_i) \qquad (2.4)$$

$$\mathbf{k}(\mathbf{x}_{N+1}) = (k(\mathbf{x}_1, \mathbf{x}_{N+1}), \ldots, k(\mathbf{x}_n, \mathbf{x}_{N+1}))^\top \qquad (2.5)$$

$$\mathbf{K} = \mathbf{\Phi}\mathbf{\Phi}^\top,\ K_{ij} = k(\mathbf{x}_i, \mathbf{x}_j) \qquad (2.6)$$

$$\mathbf{a} = (\mathbf{K} + \lambda \mathbf{I}_N)^{-1} \mathbf{y} \qquad (2.7)$$

である．また，ϵ は ϵ−insensitive 損失関数 [2, 7] における insensitive 帯域である．これはつまり，予測が，学習パターンの目標変数 y の線型結合から求められる，ということを意味している．

この式をもとに，本稿が目的としている Online SVR における逐次予測の実現を考える．ここでは，センサなどからの入力が 1 次元の場合を考え，$\dim(\mathbf{x}_i) = 1$，つまり学習データ x_i で与えられることを前提とする．基本的な考え方としては，過去のサポートベクトルから，1 時刻先の値を予測するものである．過去の l 個のサポートベクトル $x_{s_k} (k \in l)$ を用いて導かれるカーネル関数 k から構成される Gram 行列 $\mathbf{K}_{sv}$ を，

$$k(x_{s_i}, x_{s_j}) = \phi(x_{s_i})^\top \cdot \phi(x_{s_j}) = Q_{s_i, s_j} \qquad (2.8)$$

とすることで，

$$\mathbf{K}_{sv} = \begin{bmatrix} Q_{s_1,s_1} & Q_{s_1,s_2} & \cdots & Q_{s_1,s_l} \\ Q_{s_2,s_1} & Q_{s_2,s_2} & \cdots & Q_{s_2,s_l} \\ \vdots & \vdots & \ddots & \vdots \\ Q_{s_l,s_1} & Q_{s_l,s_2} & \cdots & Q_{s_l,s_l} \end{bmatrix} \qquad (2.9)$$

と定義する．また，新入力 x_{N+1} とそれまでのサポートベクトルとの特徴空間における内積を求める写像関数 $\mathbf{k}_{sv}(x_{N+1})$ については

$$\mathbf{k}_{sv}(x_{N+1}) = (k(x_{s_1}, x_{N+1}), \ldots, k(x_{s_l}, x_{N+1}))^\top \qquad (2.10)$$

とする．

以上の定義より，新入力 x_{N+1} に対する予測値 $\hat{y}_{N+1}$ を計算することを考える．

ここで, Online SVR においては, 先述のとおり学習データが追加されるに従い逐次学習を繰り返すものであった. 学習データとしてサンプル c が追加された場合, ラグランジュ乗数, バイアス項は

$$\theta' = \theta + \Delta\theta \tag{2.11}$$

$$b' = b + \Delta b \tag{2.12}$$

$$\sum_{j \in S} Q_{ij}\Delta\theta_j + \Delta b = -Q_{ic}\Delta\theta_c \qquad \text{where} \quad i \in S \tag{2.13}$$

$$\sum_{j \in S} \Delta\theta_j = -\theta_c \tag{2.14}$$

により更新される [5]. ここで, S はサポートベクトルの添え字集合 $\{s_k\}$ である. つまり, 新たなサンプル c に対するラグランジュ乗数とバイアスは, 現在までの積算値から求められる. そこで, 上式から, 予測式の構成を行う.

まず, $N = 0$ のときは, 初期状態でいかなるデータも与えられていないので,

$$\hat{y}_{N+1} = \hat{y}_1 = 0 \tag{2.15}$$

とする. つぎに $N = 1$, つまり一つ目の学習データが与えられた場合を考える. このときは x_1 以外に学習データが存在せず, 特徴空間における超平面が構成できないことから, サポートベクトル自体も存在しない. このため,

$$\Delta b = \Delta\theta \tag{2.16}$$

と導かれる. よって,

$$\hat{y}_{N+1} = \hat{y}_2 = \Delta b = \Delta\theta \tag{2.17}$$

となる. つづいて, $N > 1$ の場合を考える. このときは, x_N 以外にも学習データが存在するので, 特徴空間における超平面が構成でき, サポートベクトルが存在する. よって,

$$\hat{y}_{N+1} = \mathbf{k}_{sv}(x_{N+1})^{\top} (\mathbf{K}_{sv} + \lambda \mathbf{I}_l)^{-1} \mathbf{y}_{sv} + b' \tag{2.18}$$

となる. 以上をまとめると,

$$\hat{y}_{N+1} = \begin{cases} 0 & \text{if} \quad N = 0 \\ \Delta\theta & \text{if} \quad N = 1 \\ \mathbf{k}_{sv}(x_{N+1})^{\top} (\mathbf{K}_{sv} + \lambda \mathbf{I}_l)^{-1} \mathbf{y}_{sv} + b' & \text{otherwise} \end{cases} \tag{2.19}$$

という予測式が得られる. また, b' は, 現在までのバイアス値の積算値として求められる. この予測式を適用することで, Parrella の指摘する問題を回避し, 次状態の予測が可能となる.

3. 検証

3.1 検証で用いるパラメータ

本稿では, 検証にあたってカーネル法を適用した Online SVR を用い, カーネル関数には次式で定義される RBF カーネルを使用する.

$$k(x_1, x_2) = \exp\left(-\beta \|x_1 - x_2\|^2\right) \tag{3.1}$$

ここで, β はスケールパラメータである. また, 学習には下表 1 に示すパラメータを扱うこととする.

表 1 実験に用いる各種定数

定数	数値
λ	0.002
ϵ	0.06
β	30

3.2 検証データに適用した結果とその検討

本検証では, 予測対象の入力データとして, 周波数 1.5[Hz], 6.8[sec.] 以前は振幅 1, 6.8[sec.] 以降は振幅 0.5 の正弦波を生成し, これを 0.02[sec.] 間隔で, 0[sec.] から 11.8[sec.] までサンプリングした系列を使用した.

シミュレーションでは, このデータを逐次与え学習させるとともに, 次状態を逐次予測させた. 図 2 はそのときの結果である. 比較対象として, Parrella による, 学習終了後に安定化を実施しなかった場合の予測結果も併載する. 提案手法の結果を見ると, 学習の初期と, 傾向が変化する部分 (以下, 傾向変化部とよぶ) では, 学習データの出力を正確には予測できていないが, 時間の経過に従い, 学習データの変化に追従し, その傾向をよく表した結果となっていることがわかる. また, Parrella による予測法から得た予測結果と, 提案手法による予測結果を比較すると, 学習終了後の予測結果によく似た予測結果を, 逐次予測により得ることができた. そして安定化を要する部分でも, 不自然な変動を起こすことなく, 学習データの傾向を表すような予測が達成されていると考えられる.

また, 図 3 は, 逐次予測による予測値と, その時刻における実データとの予測誤差を示したデータである. この予測誤差 $\mathbf{e}$ は,

$$e_t = |\hat{y}_t - y_t| \tag{3.2}$$

により算出した. Parrella による予測法を用いた結果を見ると, その予測誤差は終始 0.05 程度となっているが, 0.02[sec.] で誤差が最大となっていることがわかる. これは, 学習終了後に安定化をしなかったためである. 一方, 提案手法では, 学習当初と傾向変化部では誤差が最大となっているが, その後は誤差が減少し, 2[sec.] 以降は最大でも 0.2 程度の値をとり, Parrella の予測誤差の近傍で振動していることがわかる. これは, 正弦波データの符号が変化するときとタイミングが同じであり, $y = 0$ 近傍における予測が影響しているものと考えられる. 以上から, Parrella と同等な予測結果を, 逐次的に得ることができたと考える.

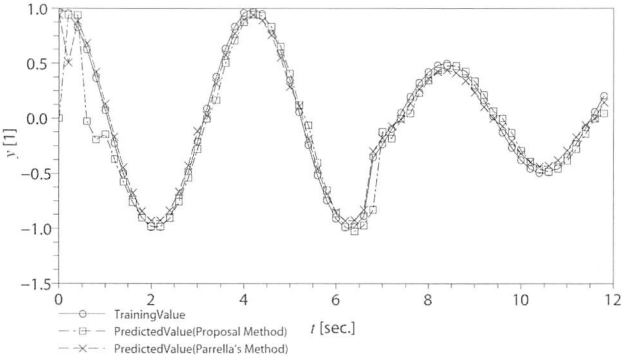


図 2 逐次予測を適用した場合の結果

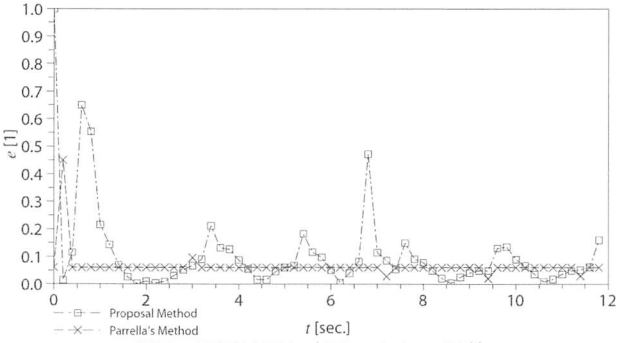


図 3 予測結果と実データとの誤差

4. 結論

本稿では, 過去のサポートベクトルと新たに入力された学習データ, そして逐次更新されるパラメータを用いて次時刻における出力の予測式を構成することで, Online SVR による逐次予測の手法について提案した. また, この提案手法を, シミュレーションにより周波数が一定の正弦波データに適用することで, 時間の経過に従い正確な予測が可能となる結果を得た. これにより, 提案手法の有用性を示した.

今後の計画としては,

- 実機に適用した場合の検証.
- ロボット制御に有用なカーネル関数の提案.
- 学習データのサンプリング間隔と回帰精度の関係についての考察.
- 学習制御系としての応用.

が挙げられる.

参考文献

[1] Corinna Cortes and V. N. Vapnik : Support-Vector Networks, Machine Learning, 20 (1995)

[2] V. N. Vapnik : The Nature of Statistical Learning Theory, Springer, New York (1995)

[3] M. Aizerman, E. Braverman, and L. Rozonoer : Theoretical foundations of the potential function method in pattern recognition learning, Automation and Remote Control 25: 821837 (1964)

[4] 小林正幸, 小西康夫, 石垣博行 : サポートベクター回帰モデルによる倒立振子の学習制御, 日本機械学会 Dynamics and Design Conference 2003, No.03-7 (2003)

[5] Francesco Parrella : Online Support Vector Regression, A Thesis presented for the degree of Information Science, Department of Information Science, University of Genoa, Italy (2007)

[6] Kuhn, H. W. and Tucker, A. W. : Nonlinear programming, Proceedings of 2nd Berkeley Symposium. Berkeley: University of California Press, pp.481-492 (1951)

[7] F. Girosi : An equivalence between sparse approximation and Support Vector Machines, Neural Compuption, 10(6):1455-1480 (1998)