総論：特集
プラントモデリングの新展開： 効率的な制御対象モデリングと制御システムの開発を目指して

機械学習技術の最近の発展とシステムモデリングへの応用

井　手　　剛＊　　矢　入　健　久＊＊

＊IBM 東京基礎研究所，神奈川県大和市下鶴間 1623-14 (LAB-S7B)
＊＊東京大学先端科学技術研究センター，東京都目黒区駒場 4-6-1
＊IBM Research – Tokyo, LAB-S7B, 1623-14 Shimo-Tsuruma, Yamato, Kanagawa 242–8502, Japan
＊＊RCAST, University of Tokyo, 4-6-1 Komaba, Meguro-ku, Tokyo 153-8904, Japan
＊E-mail: goodidea@jp.ibm.com
＊＊E-mail: yairi@space.rcast.u-tokyo.ac.jp

キーワード：機械学習, システム同定, カーネル法, Volterra 級数, 非線形性, スパース化, 動的ベイジアンネットワーク
JL 002/02/4202–0086

1. はじめに

機械学習は、「機械に自動で学習させる」という語感の通り、もともと人工知能の一分野と考えられてきた。しかし最近は、学習可能性についての数学的議論が中心であった古典的な計算論的学習理論と、統計学、多変量解析、オペレーションズリサーチ（最適化理論）、データマイニングなどの諸分野が融合し、データから有用な知見を引き出すための解析技術として大きく発展している。

元来、機械学習では、強化学習など特定の領域を除けば、独立同一分布 (i.i.d.) に従うデータを前提とした分類や回帰に関する研究が主流であり、動的なシステムに対する興味は薄かった。しかし、この状況は徐々に変わってきており、動的なシステムや時系列データ、非定常現象への興味が高まっている。例えば、現代の機械学習のバイブル的存在である Bishop の教科書[2] では、従来の教科書ではほとんど言及されていなかった隠れマルコフモデルやカルマンフィルタなどにも紙面が割かれている。このような機械学習分野の意識変化の要因としては、制御学、音声認識、信号処理、計算機視覚、ロボティクスなど、様々な動的システムを専門とする研究者が続々と参入し、相互交流が進んできたことが大きい。また、それとは逆に、制御学、特にシステム同定の分野においても、機械学習分野で発展してきた理論や技術に対する関心が高まっているようである[21]。

さて、ここ 10 年の機械学習の劇的な進歩を象徴するキーワードが、**非線形性**と**スパース性**である。前者はカーネル法とその数学的理論のことを指す。既存の学習アルゴリズムを「カーネル化（kernelize）」して、非線形性に対応させようとする試みが、2000 年前後に盛んになされた。後者については、統計学における正則化の理論を援用しつつ、たとえば L_1 正規化によるスパース化の手法が多くの実問題に適用されている。これらの研究成果は、システム同定の分野において十分に活用されているようには見えない。それを意図する研究がなされつつあるのはついここ数年のことである。本稿の目的のひとつは、主に機械学習側の観点から、そのような最新の成果を概観することである[(注1)]。

以下、次節において、機械学習の応用の具体例として、Volterra 級数の同定理論を詳しく紹介する。その後 3 節において、動的システムの同定に関する機械学習側からのアプローチを概観する。次いで 4 節においてそれに付随した研究動向を眺め、最後にまとめを述べる。

2. カーネルトリックを用いた非線形システムの同定: Volterra 級数の場合

機械学習におけるカーネル法の威力を示す具体例として、Volterra 級数展開を用いた非線形システムの同定というタスクを考える。このタスクは古典的には、cross-correlation 法という方法で行われてきたが[31]、決めるべき係数の数が多いなど実用上の問題があった。最近提案されたカーネル回帰による手法[8),9)] は、計算の精度と速度の双方において既存手法を置き換えうるものである。

2.1 Volterra 級数展開とその同定問題

非常にゆるい条件の下、一般には非線形の汎関数 $f[\cdot]$ で定義されるシステム $y(t) = f[t, u(\cdot)]$ は、次のような表現を持つ。

$$y(t) = h_0 + \sum_{n=1}^{\infty} \frac{1}{n!} \int \mathrm{d}s_1 \ \ldots \int \mathrm{d}s_n \\ h_n(s_1, ..., s_n) u(t - s_1) \ldots u(t - s_n) \quad (1)$$

これを（入力 $u(t)$ に対する）Volterra 展開と呼ぶ。この展開を 1 次までで止めた式は

$$y(t) = h_0 + \int \mathrm{d}s \ h_1(s) u(t - s)$$

であり、いわゆる線形系を表している。したがって Volterra 級数は、システムの非線形性を級数展開の形で順次取り込んだものとみなせる。

さて、時間軸を m 分割して、入力 $u(t)$ を、m 次元のベクトル $\boldsymbol{u} \in \mathbb{R}^m$ とみなすことにする。すると引数を適当

(注1) 動的システムを扱い、かつ、制御学とも関連の深い話題として強化学習[29] があるが、紙幅の制約もあり、本稿では扱わない。

に読みかえることで、次のような離散 Volterra 展開が得られる。

$$y = h^0 + \sum_{n=1}^{\infty} \sum_{i_1=1}^{m} \cdots \sum_{i_n=1}^{m} h^n_{i_1,...,i_n} u_{i_1} \cdots u_{i_n} \qquad (2)$$

ただし、定常なシステムを仮定して、Volterra 係数 $h^n_{i_1,...,i_n}$ から時刻を表す添え字 t を省いた。

この表現におけるシステム同定問題を改めて形式的に述べておく。Volterra 級数表現におけるシステム同定とは、データ

$$\mathcal{D} \equiv \left\{ (\boldsymbol{u}^{(t)}, y^{(t)}) \,\middle|\, \boldsymbol{u}^{(t)} \in \mathbb{R}^m,\ \ y^{(t)} \in \mathbb{R}^1,\ \ t = 1, 2, ..., N \right\}$$

が与えられたときに、Volterra 係数

$$h^0, \{h^1_{i_1}\}, ..., \{h^n_{i_1,...,i_n}\}, ...$$

を定めることである。係数は一般に無限個あるので、従来手法による同定は原理的な困難を抱えている。しかし以下に見るように、カーネル法は、この無限個をカーネル関数の中に「繰り込む」ことを可能にするのである。

2.2 Volterra 級数のベクトル表現

今、Volterra 級数の各項に対応して、m 個の変数から作られる n 次の単項式を要素とするベクトルを考える。まず $\boldsymbol{\phi}^0(\boldsymbol{u}) \equiv 1$ と定義する。1 次であれば

$$\boldsymbol{\phi}^1(\boldsymbol{u}) \equiv [u_1, ..., u_m]^\top \ \in \mathbb{R}^m$$

であり、2 次であれば

$$\boldsymbol{\phi}^2(\boldsymbol{u}) \equiv [u_1 u_1, u_1 u_2, ..., u_m u_m]^\top \ \in \mathbb{R}^{m^2} \qquad (3)$$

である。言うまでもなくこの写像 $\boldsymbol{u} \to \boldsymbol{\phi}^n(\boldsymbol{u})$ は、もともと m 次元のベクトルを、m^n という高次元空間に移すもので、非常に冗長な表現である。しかしとにかくこういう「特徴ベクトル」を使うと離散 Volterra 展開の式が、

$$y = \sum_{n=0}^{\infty} \boldsymbol{\eta}_n^\top \boldsymbol{\phi}^n(\boldsymbol{u})$$

と書ける。ここで、$\eta_0 = h^0$ であり、$\boldsymbol{\phi}^1$ および $\boldsymbol{\phi}^2$ に対応して、

$$\boldsymbol{\eta}_1 \equiv [h^1_1, h^1_2, ..., h^1_m]^\top \ \in \mathbb{R}^m \qquad (4)$$

$$\boldsymbol{\eta}_2 \equiv [h^1_{1,1}, h^1_{2,1}, ..., h^1_{m,m}]^\top \ \in \mathbb{R}^{m^2} \qquad (5)$$

などである。上式は、未知の Volterra 係数からなるベクトル $\{\boldsymbol{\eta}_n\}$ と、特徴ベクトルとの内積の和として書けることを意味している。さらに、係数ベクトルと特徴ベクトルを縦に積み重ねて、形式的に無限次元のベクトル

$$\boldsymbol{\eta} \equiv \left[\eta_0, \boldsymbol{\eta}_1^\top, \boldsymbol{\eta}_2^\top, ..., \boldsymbol{\eta}_\infty^\top\right]^\top \qquad (6)$$

$$\boldsymbol{\phi}(\boldsymbol{u}) \equiv \left[1, \boldsymbol{\phi}_1^\top, \boldsymbol{\phi}_2^\top, ..., \boldsymbol{\phi}_\infty^\top\right]^\top \qquad (7)$$

を定義する。これを使うと、Volterra 級数を簡潔に

$$y = \boldsymbol{\eta}^\top \boldsymbol{\phi}(\boldsymbol{u})$$

と書ける。この表現においては、システム同定問題とは、データ $\mathcal{D}$ から $\boldsymbol{\eta}$ を定めることである。

2.3 特徴ベクトル同士の内積の計算

離散 Volterra 展開から上記のように導かれた非線形写像 $\boldsymbol{\phi}^n$ は次のような性質を持っている。

$$\begin{aligned} \boldsymbol{\phi}^n(\boldsymbol{u})^\top \boldsymbol{\phi}^n(\boldsymbol{z}) &= \sum_{i_1=1}^{m} \cdots \sum_{i_n=1}^{m} u_{i_1} \cdots u_{i_n} z_{i_1} \cdots z_{i_n} \\ &= \sum_{i_1=1}^{m} \cdots \sum_{i_n=1}^{m} (u_{i_1} z_{i_1}) \cdots (u_{i_n} z_{i_n}) = \left(\boldsymbol{u}^\top \boldsymbol{z}\right)^n \end{aligned}$$

今、$\boldsymbol{\phi}$ の定義を拡張して、$\{\beta_1, \beta_2, ... \in \mathbb{R}\}$ に対して

$$\boldsymbol{\phi}(\boldsymbol{u}) \equiv \left[\beta_0 \boldsymbol{\phi}^0(\boldsymbol{u})^\top, \beta_1 \boldsymbol{\phi}^1(\boldsymbol{u})^\top, ... \beta_\infty \boldsymbol{\phi}^\infty(\boldsymbol{u})^\top\right]^\top$$

としてみよう。この時、特徴ベクトル $\boldsymbol{\phi}$ の内積は、

$$k(\boldsymbol{u}, \boldsymbol{z}) \equiv \boldsymbol{\phi}(\boldsymbol{u})^\top \boldsymbol{\phi}(\boldsymbol{z}) = \sum_{n=0}^{\infty} {\beta_n}^2 \left(\boldsymbol{u}^\top \boldsymbol{z}\right)^n \qquad (8)$$

のように計算できる。これは無限級数であり、計算可能であるかどうかは直ちには明らかでない。そこで発想をやや変えて、上式が容易に計算できるように、係数 $\{{\beta_n}^2\}$ を調整することを考える。たとえば、p までの係数を 2 項係数に選び、以降を 0 とすれば、

$$k(\boldsymbol{u}, \boldsymbol{z}) = \left(1 + \boldsymbol{u}^\top \boldsymbol{z}\right)^p \qquad (9)$$

となる。また、${\beta_n}^2 = \frac{1}{n!}$ とすれば

$$k(\boldsymbol{u}, \boldsymbol{z}) = \exp\left(\boldsymbol{u}^\top \boldsymbol{z}\right) \qquad (10)$$

となることがわかる。これらは、無限次元のよくわからない特徴ベクトルを直接触ることなく、元の入力 $\boldsymbol{u}$ から直ちに計算できる量である。上記の $k(\boldsymbol{u}, \boldsymbol{z})$ は、元の入力を引数にして、特徴ベクトル同士の内積を返す関数であるが、このような関数を**カーネル関数**と呼ぶ。そうして、データ $\mathcal{D}$ における N 個のサンプルから、カーネル関数を介して作られる $N \times N$ 行列 $\mathsf{K} \equiv (k(\boldsymbol{u}^{(i)}, \boldsymbol{u}^{(j)}))$ を**カーネル行列**と呼ぶ。

2.4 カーネルリッジ回帰によるシステム同定

さて、先に述べたように、われわれの問題は、データ $\mathcal{D}$ から Volterra 係数を定めることである。あるいはより一般的に考えれば、任意の $\boldsymbol{u}$ に対して y を与える予測式を構築することである。この問題を解くために、従来の制御理論でもよく行われてきたように、2 乗誤差を最小化することを考える。すなわち、Volterra 係数 $\boldsymbol{\eta}$ を

$$\Psi(\boldsymbol{\eta}|\lambda) \equiv \frac{1}{N} \sum_{n=1}^{N} |y^{(n)} - \boldsymbol{\eta}^\top \boldsymbol{\phi}(\boldsymbol{u}^{(n)})|^2 + \lambda ||\boldsymbol{\eta}||^2 \qquad (11)$$

を最小化するように定める。上式において、λ はデータから決められるある定数であるが、さしあたり所与のものとする。この第 2 項はいわゆる L_2 正規化項と呼ばれるものであり、詳しい議論は省くが、解を安定化させる働きを持つ。$\lambda = 0$ とすれば通常の最小 2 乗法である。

係数ベクトル $\boldsymbol{\eta}$ はもともと無限次元を持つものとして定義されていた。今、その代わりに、

$$\boldsymbol{\eta} \equiv \Phi\boldsymbol{\alpha} \tag{12}$$

を満たす N 次元ベクトル $\boldsymbol{\alpha}$ を考える。ここで Φ は

$$\Phi \equiv \left[\boldsymbol{\phi}(\boldsymbol{u}^{(1)}), \boldsymbol{\phi}(\boldsymbol{u}^{(2)}), ..., \boldsymbol{\phi}(\boldsymbol{u}^{(N)})\right] \in \mathbb{R}^{\infty\times N}$$

という行列である。これを使って目的関数 Ψ を書き直すと、今度は $\boldsymbol{\alpha}$ の関数として

$$\Psi(\boldsymbol{\alpha}|\lambda) \equiv \frac{1}{N}||\boldsymbol{y}_N - \mathsf{K}\boldsymbol{\alpha}||^2 + \lambda\boldsymbol{\alpha}^\top\mathsf{K}\boldsymbol{\alpha} \tag{13}$$

となる。ただし、$\boldsymbol{y}_N \equiv \left[y^{(1)}, y^{(2)}, ..., y^{(N)}\right]^\top$ と定義した。上記の目的関数を最小化する回帰問題を、カーネルリッジ回帰（kernel ridge regression）と呼ぶ。

目的関数 (13) を最小化する $\boldsymbol{\alpha}$ を求めるのは容易であり、初等的なベクトルの微分公式を適用することで最適解が

$$\boldsymbol{\alpha}^* = \left[\mathsf{K} + \lambda N\mathsf{I}_N\right]^{-1}\boldsymbol{y}_N$$

と求まる。ここで I_N は N 次元の単位行列である。これを使えば、式 (12) より、任意の入力 $\boldsymbol{u}$ に対する出力が

$$y = \boldsymbol{\phi}(\boldsymbol{u})^\top\Phi\boldsymbol{\alpha}^* = \boldsymbol{k}(\boldsymbol{u})^\top\left[\mathsf{K} + \lambda N\mathsf{I}_N\right]^{-1}\boldsymbol{y}_N \tag{14}$$

と求められる。ただし、

$$\boldsymbol{k}(\boldsymbol{u}) \equiv \left[k(\boldsymbol{u}, \boldsymbol{u}^{(1)}), k(\boldsymbol{u}, \boldsymbol{u}^{(2)}), ..., k(\boldsymbol{u}, \boldsymbol{u}^{(N)})\right]^\top \in \mathbb{R}^N$$

である。これでシステム同定ができたことになる。結局、解は、カーネル関数 k を通してのみデータに依存する。上記の解には N 次元ベクトルおよび $N \times N$ 次元行列しか出てこない。Volterra 級数の無限個の係数は K の中に繰り込まれてしまったわけである。

式をたどると、このマジックの由来は、式 (12) にある。これは無限次元の係数ベクトル $\boldsymbol{\eta}$ を、N 次元の係数ベクトル $\boldsymbol{\alpha}$ で表現するという式であるが、やや驚くべきことに、カーネル法の基本定理である Representer 定理 [33] によれば、この変換により何の一般性も失われないことが示せるのである。このような変換とそれによる無限次元の繰り込みを**カーネル・トリック**と呼ぶ。

3. 機械学習に基づく動的システムの同定技術の研究動向

前節では、カーネルトリックという手法により、無限個の係数を持つ Volterra 級数が、高々データの個数 N の次元を持つ行列の計算に帰着されることを見た。本節では、より一般的な見地から、動的システムの解析に関する機械学習の諸手法の最新動向について概観する。

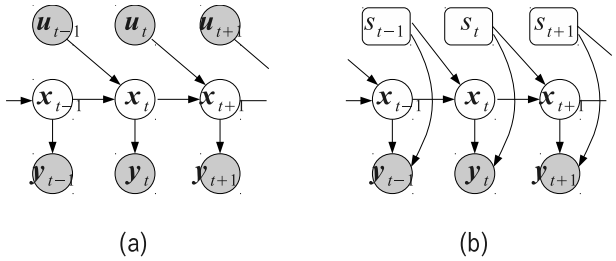

図 1　DBN の例: (a) 状態空間モデル (入力あり), (b) スイッチングモデル

詳細に入る前に、機械学習分野で動的システムを扱う際によく用いられる**動的ベイジアンネットワーク** (Dynamic Bayesian Networks: DBN)[23] を紹介しておこう。ベイジアンネットワーク (Bayesian Networks: BN) は、変数間の確率的な依存・独立関係を有向非巡回グラフによって表現するモデル化法であり、90 年代以降の人工知能分野において盛んに研究されてきた。その BN を、対象システムの時間発展を記述できるように拡張したのが DBN である。例えば、離散時間状態空間モデル

$$\boldsymbol{x}_{t+1} = f(\boldsymbol{x}_t, \boldsymbol{u}_t) + \boldsymbol{v}_t \tag{15}$$

$$\boldsymbol{y}_t = g(\boldsymbol{x}_t) + \boldsymbol{w}_t \tag{16}$$

を DBN で表現すると図 1 (a) のようになる。ここで、直接観測することのできない状態変数 $\boldsymbol{x}_t, \boldsymbol{x}_{t+1}$ は、隠れ変数もしくは潜在変数などと呼ばれる[(注2)][(注3)]。また、$\boldsymbol{v}_t, \boldsymbol{w}_t$ はノイズを表す。

DBN は離散・連続変数の混在を許容するなど、非常に柔軟性の高いモデル化法である。特に機械学習の文脈では、モデルの学習（上式でいえば f と g の関数形の同定）、および、モデル構造の学習（図 1 で言えばネットワークのトポロジー自体の学習）について多くの研究がなされている[10),11),23)]。近年では、時間的にモデル構造やパラメータが変化するような、非定常 DBN の学習に関心が集まっている[30),34)]。

以下、少し対象を限定し、連続的な状態空間を持つ非線形な動的システムの学習・同定問題に対して、非線形回帰や次元削減などの機械学習手法がどのように利用されてきたかを概観することにしよう。

3.1　スイッチング線形モデルの学習

未知の非線形システムをモデル化する最も基本的かつ現実的な方法は、複数の線形モデルによって局所的に近似する

(注2) DBN ではしばしば、観測できる変数 (この場合は $\boldsymbol{u}_t$,$\boldsymbol{y}_t$) を表すノードを塗り潰して、直接観測できない変数 (この場合は $\boldsymbol{x}_t$) と区別することがある。

(注3) フィードバックが存在する場合も、$\boldsymbol{x}_t$ から $\boldsymbol{u}_t$ への矢印を加えれば同様に扱える。

ことであろう[24]。機械学習の分野では、そのようなモデルは、スイッチング線形動的システム (Switching Linear Dynamical System: SLDS)、あるいはスイッチングカルマンフィルタ (SKF) などと呼ばれており、先の図 1 (b) のような DBN による表現が可能である。

SLDS モデルをデータから学習するには、通常、EM (Expectation-Maximization) アルゴリズムや、それに類する反復的アルゴリズムが用いられる。すなわち、まず、E ステップでは、モデルを固定して確率推論を行い、2 種類の隠れ変数、すなわち、s_t(モード変数:どの局所モデルを選択するかを表す離散的確率変数) と $\boldsymbol{x}_t$(連続的な状態変数) に関する確率密度分布を推定する。次に、M ステップでは、E ステップで得られた s_t と $\boldsymbol{x}_t$ の分布を用いて、尤度を最大化するモデルを求める。

このうち、M ステップについては、各線形モデルが最小二乗法などにより容易に推定できる。一方、E ステップについては、隠れ変数 s_t と $\boldsymbol{x}_t$ の同時事後分布を厳密に推定することが困難であるため、何らかの近似推定法を用いる必要がある。例えば文献[25] では、s_t の推定に Viterbi アルゴリズムを用いる方法や、変分近似を用いる方法[12] などが比較されている。

また、SLDS の学習では、最適な局所線形モデル数の選択という難問題が存在するが、最近の研究[4),7)] では、ディリクレ過程混合モデルや変分ベイズ法を用いることによってモデル数を自動決定する手法が提案されている。

3.2 非線形潜在変数モデルの学習

非線形システムを学習するもう一つのアプローチは、状態空間モデル (15) および (16) において、状態遷移関数 f および 出力関数 g を、非線形回帰や次元削減の手法を用いて観測データから推定することである。

（1） 動径基底関数ネットワークモデル

まず、カーネル以前の非線形アプローチの例としては、Ghahramani と Roweis による研究[13] が有名である。これは、状態空間モデルにおける状態方程式および観測方程式を Radial Basis Function Networks (RBFN) によって表現し、そのモデルパラメータを EM アルゴリズムによって反復的に学習するというものである。つまり、E ステップにおいては、現在までに推定されているモデルを用いて拡張カルマンスムージング (Extended Kalman Smoothing) を行い状態列を推定する。そして、M ステップでは推定された状態列を用いて RBFN のパラメータを学習する。

（2） カーネル部分空間モデル

カーネルによる非線形特徴空間写像を利用した動的システムの学習については、これまでにいくつかの方式が提案されているが、学習のどの部分でカーネルを使うかという点にバリエーションがあり興味深い。

Kernel Kalman Filter (KKF) [27] は、「非線形なシステムであっても、カーネルによって写像された特徴空間では線形なモデルで表される」という考えに基づき、線形の状態空間モデルをカーネルトリックによって非線形化したものである。この手法では、特徴空間における状態・観測方程式のモデルパラメータはやはり EM アルゴリズムによって学習される。

他方、文献[14] は、部分空間同定法とカーネルリッジ回帰を用いて、Hammerstein 型の非線形システムの同定を行っている。また、カーネルと部分空間同定法を融合した他の例としては、カーネル正準相関分析 (Kernel Canonical Correlation Analysis: KCCA) を用いた Kawahara らの研究[17] が挙げられる。

（3） 正規過程モデル

ところで、カーネル行列を分散共分散行列とみなすことによって**正規過程** (Gaussian Process)[28] と呼ばれる確率過程が導かれるが、これを動的システムに応用した手法として、正規過程動的モデル (Gaussian Process Dynamical Models: GPDM)[35] がある。この方法のユニークな点は、非線形の状態方式 f, 観測方程式 g を直接推定するのではなくて積分消去してしまい、隠れ状態列とその分散共分散行列を事後確率最大化によって求めるところである。ただし、実際の計算は、勾配法的な数値最適化によるため、計算コストと局所解が問題になる。

他にも、状態遷移関数 f, 出力関数 g を正規過程によってモデル化するものとして、GP-Bayes[18] や GP-ADF[5] などがある。ただし、これらは訓練時において状態変数 $\{\boldsymbol{x}_t\}$ の値が与えられると仮定しており、典型的な教師あり回帰学習の問題に帰着させている。

（4） 多様体学習モデル

ところで近年、カーネルとも関連して、いわゆる多様体学習 (Manifold Learning) と呼ばれる非線形次元削減技術が注目されている (例えば、文献[32] 参照)。これまでのところ、計算コストの問題などから多様体学習の非線形動的システムへの応用は多くないが、ラプラス固有写像 (Laplacian Eigenmap:LE)[1] を確率的な潜在変数モデルに拡張し、非線形な観測モデルとして利用した例[22] がある。ただし、この例では状態遷移には単純なランダムウォークモデルを利用している。

4. その他の研究

ここではシステム同定に付随して現れる研究課題の研究動向を概観する。

4.1 カーネル法の近似理論

カーネル法はもとの最適化問題の双対問題を解くことで、問題を、サンプル数 N の項を持つカーネル展開の係数を決める作業に帰着させる。その結果、モデルの次数に関する「次元の呪い」を避けることを可能にする。しかしその結果現れる最適化問題は、$N \times N$ 行列の 1 次方程式や固有値問題を含み、サンプル数が大きくなると計算量的な問題が生

じる。これは一般にカーネル法における深刻な問題であって、機械学習の分野でもいまだに活発な研究が行われている文[16]。

システム同定の分野でも、最近この困難が意識されるようになってきた。De Moor のグループでは、非線形システム同定問題にカーネルリッジ回帰と Nyström 近似を併用するアプローチを提案している[3),6)]。Nyström 近似というのは、カーネル行列 K を、データのランダムサンプリングで選んだ少ないサンプルを使って、より小さいサイズの行列で近似する手法である[37)]。Nyström 近似は一見魅力的であるが、実は深刻な不安定性があることが実験的に明らかにされており[36)]、使用には注意を要する。

4.2 スパース化技術の応用

Pelckmans ら[26)]は、非線形の ARX モデルをカーネル回帰の方法で解くにあたり、リッジ回帰の「自由度」という量をモデルの複雑さの指標として使うことを提案している。回帰問題の自由度とは、入力 $\boldsymbol{u}$ の形式的な次元 M のうち、出力の予測に本当に寄与している次元の数に対応するもので、元は統計学の分野で考案されたものである[15)]。Pelckmans らはまた、目的関数 (11) の第 2 項を、いわゆる L_1 正則化項に置き換えることにより、スパースな解を得る試みをしている。一般に 2 乗誤差項に L_1 正則化項を加えて解く回帰問題を、Lasso (least absolute shrinkage and selection operator) と呼ぶ。この場合、システムの自由度は、非ゼロの係数の個数で与えられる。

同じく、Kukreja らは[19)]、非線形システムの同定問題を Lasso を用いて解き、データの SN 比が高い場合には精度と簡潔さを両立させたモデルを求められることを実験的に示した。

これらの研究は、統計学や機械学習の分野で研究されてきた解のスパース化というコンセプトをシステム同定の分野にいち早く応用したものとして注目される。解のスパース化は、特徴選択と密接なかかわりを持つ。Lasso 回帰においては、λ の値の選択にもよるが、一般に多くの係数 η_i が厳密に 0 になる。言い換えると、出力 y をあまり説明しない特徴は、自動的にモデルから除外される。実用的なプラントモデリングで重要になる複合系においては、「どの部分系が本質的か」という情報は非常に有用である。このような「割り切り」の用途に、スパース化という技術は重要になると思われる。ただし、Lasso の定式化では、Representer 定理というカーネル法の基本定理[33)]が成立せず、数学的に首尾一貫した方法で非線形性を取り込むのが難しいことに注意したい。

4.3 サポートベクトル回帰

近年の機械学習を代表するアルゴリズムと言えば、多くの読者はサポートベクトルマシン (SVM) を挙げるのではなかろうか。しかし、非線形動的システムの学習に SVM の回帰学習バージョンである SVR（support vector regression）を利用した例は意外に少ない。その理由は、基本的に SVM(SVR) は確率的な手法でないため、ベイズの定理に基づく確率的推論との相性が悪いからである。そのため、前述のように、正規過程回帰などの確率的な非線形回帰手法が好まれて利用されている。

しかし最近、この風潮を変える可能性のある研究が Langford らによって発表されている[20)]。その基本的なアイデアは、確率的な状態変数をそのまま扱う代わりに、その十分統計量を決定論的な状態変数として利用する、というなものである。これにより、前述の SVM を含めて、任意の (確率的でない) 非線形回帰アルゴリズムを、確率的な動的システムの学習に用いることができるようになるという。

本研究に象徴されるように、機械学習の側からも、動的システムの学習というテーマはまだ発展途上である。今後、制御学との相互作用を通して、実世界にインパクトを持つ研究が数多く現れることを期待したい。

5. むすび

ここ 10 年の機械学習の技術の進歩を象徴するキーワードが、非線形性とスパース性である。本稿では主に前者、カーネル法のシステム同定への適用を中心に、最新の研究動向を概観した。非線形性の取り込みは、機械学習の分野ではもはや特別なことではない。しかしその一方、計算量など、別の研究課題も生ずることを述べた。

機械学習のもうひとつの大きな成果であるスパース化の技法に関しては、機械学習の分野では、特徴選択ないし代表サンプルの選択のための標準的な手法としての地位を確立している[2)]。L_1 正則化は制御工学の一部でも周知の技術であるが、機械学習の分野における実問題で培われた様々な問題設定や技法が、今後、プラントモデリングに刺激を与える可能性が大いにあると思われる。

参 考 文 献

1） M. Belkin and P. Niyogi. Laplacian eigenmaps and spectral techniques for embedding and clustering. In *Advances in Neural Information Processing Systems 14*, pages 585–591. MIT Press, 2001.

2） C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

3） K. D. Brabanter, P. Dreesen, P. Karsmakers, K. Pelckmans, J. D. Brabanter, J. Suykens, and B. D. Moor. Fixed-size LS-SVM applied to the Wiener-Hammerstein benchmark. In *Proceedings of the 15th IFAC Symposium on System Identification*, 2009.

4） S. Chiappa, J. Kober, and J. Peters. Using Bayesian dynamical systems for motion template libraries. In *Advances in Neural Information Processing Systems 21*. MIT Press, 2009.

5） M. P. Deisenroth, M. F. Huber, and U. D. Hanebeck. Analytic moment-based Gaussian process filtering. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 225–232, 2009.

6） T. Falck, K. Pelckmans, J. Suykens, and B. D. Moor. Identification of Wiener-Hammerstein systems using LS-SVMs. In *Proceedings of the 15th IFAC Symposium on System Identification*, pages 820–825, 2009.

7） E. Fox, E. Sudderth, M. Jordan, and A. Willsky. Nonparametric Bayesian learning of switching linear dynamical systems. In *Advances in Neural Information Processing Systems 21*. MIT Press, 2009.
8） M. O. Franz and B. Schölkopf. Implicit Wiener series for higher-order image analysis. In L. K. Saul, Y. Weiss, and L. Bottou, editors, *Advances in Neural Information Processing Systems 17*, pages 465–472, Cambridge, MA, 2005. MIT Press.
9） M. O. Franz and B. Schölkopf. A unifying view of Wiener and Volterra theory and polynomial kernel regression. *Neural Computation*, 18(12):3097–3118, 2006.
10） N. Friedman, K. Murphy, and S. Russell. Learning the structure of dynamic probabilistic networks. In *Proc. Fourteenth Conference on Uncertainty in Artificial Intelligence (UAI-98)*, pages 139–147, 1998.
11） Z. Ghahramani. Learning dynamic Bayesian networks. In *Adaptive Processing of Sequences and Data Structures (Lecture notes in artificial intelligence)*, pages 168–197. Springer-Verlag, 1997.
12） Z. Ghahramani and G. E. Hinton. Variational learning for switching state-space models. *Neural Computation*, 12:963–996, 1998.
13） Z. Ghahramani and S. Roweis. Learning nonlinear dynamical systems using an EM algorithm. In *Advances in Neural Information Processing Systems 11*, pages 431–437. MIT Press, 1999.
14） I. Goethals, K. Pelckmans, J. A. K. Suykens, and B. D. Moor. Subspace identification of Hammerstein systems using least squares support vector machines. *IEEE Trans. on Automatic Control*, 50:1509–1519, 2005.
15） T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning — Data Mining, Inference, and Prediction*. Springer, 2001.
16） H. Kashima, T. Idé, T. Kato, and M. Sugiyama. Recent advances and trends in large-scale kernel methods. *Transactions on Information and Systems*, E92-D(7):1338–1353, 2009.
17） Y. Kawahara, T. Yairi, and K. Machida. A kernel subspace method by stochastic realization for learning nonlinear dynamical systems. In B. Schölkopf, J. C. Platt, and T. Hofmann, editors, *Advances in Neural Information Processing Systems 19*, pages 665–672, Cambridge, MA, 2007. MIT Press.
18） J. Ko and D. Fox. Gp-bayesfilters: Bayesian filtering using Gaussian process prediction and observation models. *Autonomous Robots*, 27(1):75–90, 2009.
19） S. L. Kukreja, J. Löfberg, and M. J. Brenner. A least absolute shrinkage and selection operator (LASSO) for nonlinear system identification. In *Proceedings of the14th IFAC Symposium on System Identification*, volume 14.
20） J. Langford, R. Salakhutdinov, and T. Zhang. Learning nonlinear dynamic models. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 593–600, 2009.
21） L. Ljung. Perspectives on system identification. In *the IFAC Congress*, 2008.
22） Z. Lu, M. Carreira-Perpinan, and C. Sminchisescu. People tracking with the Laplacian eigenmaps latent variable model. In J. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems 20*, pages 1705–1712. MIT Press, Cambridge, MA, 2008.
23） K. P. Murphy. *Dynamic Bayesian Networks: Representation, Inference and Learning*. PhD thesis, UC Berkeley, 2002.
24） R. Murray-Smith and T. A. Johansen. *Multiple Model Approaches to Modelling and Control*. CRC Press, 1997.
25） V. Pavlovic, J. M. Rehg, and J. Maccormick. Learning switching linear models of human motion. In *Advances in Neural Information Processing Systems 13*, pages 981–987. The MIT Press, 2001.
26） K. Pelckmans, K. Goethals, J. Suykens, and B. D. Moor. On model complexity control in identification of Hammerstein systems. In *Proceedings of IEEE Conference on Decision and Control*, volume 2, pages 1203–1208, 2005.
27） L. Ralaivola and F. d'Alche Buc. Time series filtering, smoothing and learning using the kernel Kalman filter. In *Proc. of IEEE Int. Joint Conference on Neural Networks*, pages 1449–1454, 2005.
28） C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006.
29） A. G. B. Richard S. Sutton. *Reinforcement Learning: An Introduction*. The MIT Press, 1998.
30） J. W. Robinson and A. J. Hartemink. Non-stationary dynamic Bayesian networks. In D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, editors, *Advances in Neural Information Processing Systems 21*, pages 1369–1376. MIT Press, 2009.
31） W. J. Rugh. *Nonlinear System Theory — The Volterra/Wiener Approach*. The Johns Hopkins University Press, 1981. (Web version prepared in 2002.).
32） L. K. Saul, K. Q. Weinberger, J. H. Ham, F. Sha, and D. Lee. Chapter 16: Spectral methods for dimensionality reduction. In *Semisupervised learning*, pages 279–294. MIT Press, 2006.
33） B. Schölkopf and A. J. Smola. *Learning with Kernels*. The MIT Press, 2002.
34） L. Song, M. Kolar, and E. Xing. Time-varying dynamic Bayesian networks. In Y. Bengio, D. Schuurmans, J. Lafferty, C. K. I. Williams, and A. Culotta, editors, *Advances in Neural Information Processing Systems 22*, pages 1732–1740. MIT Press, 2009.
35） J. Wang, D. Fleet, and A. Hertzmann. Gaussian process dynamical models. In *Advances in Neural Information Processing Systems 18*, pages 1441–1448. MIT Press, 2006.
36） C. K. I. Williams, C. E. Rasmussen, A. Schwaighofer, and V. Tresp. Observations on the Nyström method for Gaussian process prediction, `http://www.dai.ed.ac.uk/homes/ckiw/online_pubs.html`. 2002.
37） C. K. I. Williams and M. Seeger. Using the Nyström method to speed up kernel machines. In *Advances in Neural Information Processing Systems 13*, pages 682–688, 2001.

［著　者　紹　介］

井手剛（いで つよし）君(非会員)

国立苫小牧工業高等専門学校機械工学科、東北大学工学部機械工学科を経て、2000 年東京大学大学院理学系研究科・物理学専攻博士課程修了。同年 IBM 東京基礎研究所入所。現在、同研究所アドバイザリー・リサーチャー。機械学習・データマイニングの研究に従事。人工知能学会、電子情報通信学会、日本物理学会、ACM SIGKDD 各会員。

矢入健久（やいり たけひさ）君(非会員)

東京大学工学部航空学科卒業。1999 年同大学院工学系研究科航空宇宙工学専攻博士課程修了。同大学先端科学技術研究センター助手、講師などを経て、2006 年より同大学工学系研究科准教授。博士（工学）。機械学習，移動ロボットの環境モデル獲得，システム異常検知・診断などの研究に従事。人工知能学会、日本ロボット学会、日本航空宇宙学会各会員。