

二重マッピングとSCWによる株価変動予測

Predicting stock fluctuations using Two-level Mapping and SCW

福田 ムフタル^{1*}
Muhtar Fukuda¹¹ 名古屋産業大学環境情報ビジネス学部¹ Faculty of Environmental and Information Studies, Nagoya Sangyo University

Abstract: Due to high uncertainty in the stock market, it is difficult to predict the future fluctuations of stock prices even if we use the state-of-the-art techniques of machine learning, such as Deep Learning. However, in some cases with choosing an appropriate machine learning algorithm, feature values and outputs for the prediction, we can have desirable predicted results, especially on short-term stock fluctuations about some market indices. Some initial reliable results have been achieved in our related work, by using Soft Confidence-Weighted (SCW) Learning, which is one of online learning. In this paper, we propose a predicting method using two-level mapping and SCW. We will focus on feature transformations using the two-level mapping. The first one is based on the mathematical concept of the Singular Value Decomposition (SVD), to get strong convergence and higher accuracy. The second one is to make the predicted Fluctuation Strength (FS) more precisely, in which we use pre-learned outputs and do relearning.

1 はじめに

株式市場には、常に需要と供給があり、そのバランスによって市場価格が形成され、またはその価格で取引されると今までは考えられてきた。そして投資家たちはそれぞれの判断基準に基づいて株式投資を行ってきた。しかし、ICT技術を基盤とした株式投資環境の発展に伴い、個人投資家を含めた市場参加者の拡大、マーケットのグローバル化、SNS等を手段としたソーシャルメディアネットワークの普及により、株価形成に影響を与える要素が複雑化し、不確実性も増大した。そういったことから市場動向を伝統的な経済学や金融工学の理論だけでは説明できないケースが増え、一定以上の投資パフォーマンスの確保が一層困難になった。株価変動はランダム・ウォークであると言われる場合がある。そうであれば一定の確率分布に従うはずであり、何らかの理論的な説明ができる。しかし、最近の変動はランダムどころか、何重にも攪乱に遭い、本来の姿が消えてさえているとも思える。

上で述べたように、説明しにくくなった金融市場において、経験の浅い個人投資家だけではなく、プロの機関投資家にとっても如何にして利益の最大化とリスクの最小化を実現できるかがこれまで以上に重要になってきた。一方、人工知能の分野では前述の状況と並行し

て、本来の姿が表面的には見えないが、大量に蓄積された過去のデータから潜在的な相関関係を見出し、それらに対して機械学習を行い、様々な予測や判断に利用できる機械学習手法が考案され、著しい成果が上がっている。その中で特に注目を浴びているのがディープラーニング(DL)[1]であろう。

ディープラーニングは人間の脳の働きをモデル化したものの一つであり、ベテランの金融アナリストや投資家たちが過去の市場動向や長年の投資経験から多くを学んだのと同様に学習させ、投資判断に応用できると期待されている。その応用例として Deep Belief Network(DBN)[2]を用いた為替レート変動の予測[3]や上場企業倒産の予測[4]等の研究が報告され、ある程度の有効性が示されている。しかし、多階層のネットワーク構造であるDLを用いて金融時系列データの学習を行う場合、依存性の高いパラメータが多数あり、その値を微調整するだけで収束性と予測結果が激変するし、かなりの学習時間も要する。金融時系列データには時間的従属関係があり、学習期間と学習のための入力データである属性値(feature)の構成要素を常に最新のものに更新して再学習を繰り返す必要があり、更新のたびに収束性と予測結果に大きなばらつきが生じると、予測結果の信頼性が揺らぐことになる。

学習の安定的な収束とよい予測結果が期待できるもう一つの機械学習モデルとして Support Vector Machine (SVM)[5]があり、SVMによる高精度のトレンド予測

*連絡先：名古屋産業大学環境情報ビジネス学部
〒488-8711 愛知県尾張旭市新居町山の田 3255-5)
E-mail: fukuda@nagoya-su.ac.jp

を実現した研究 [6, 7] もある。しかし、SVM を用いて金融時系列データの学習と予測を行う場合、その正解率が学習データのサイズと属性値の選択に依存するところが大きい [7]。また、金融時系列データは非線形であり、SVM モデルの予測正解率を上げるためには、非線形カーネルを使用する必要がある。非線形カーネルにはやはり複数のパラメータがあり、SVM の振る舞いにかなり影響を及ぼすため、その調整を慎重にかつ時間をかけて行わなければならない、学習期間と属性値を常に更新する機械学習では問題となる。

また、以上の各モデルを含め、従来の予測モデルの殆どは翌日、あるいは次の一定期間の株価変動の方向性一致率に重点をおいており、ある銘柄の次の日の株価を「上がる」と予測し、実際の株価が例えば 0.01% でも上がった場合、それが正解とされる。それは理論的には正解となっても、実際の投資目的には不十分である。株価予測の場合、株価が上がるか下がるかだけではなく、その騰落の程度を定量的に示すことがより重要である。買って売ってもコストに見合ったリターンが望めない場合は何もせず、一定以上のリターンが望めるタイミングで売買を行うのが実用的である。

本研究では、より最近の過去データが次の株価変動の予測により大きな影響を与え、かつ近隣の学習データができるだけ独立であるべきだと考えており、ディープラーニングも SVM も基本的にはバッチ学習であるため、本研究の考え方にはそぐわない。

著者は先行研究として、以上のような諸問題を避けつつ、株価が上がるか下がるかの方向性だけではなく、株式の売買タイミングにも重点をおいた株価変動予測の一手法 [9] を提案し、一定の有効性を示した。その中では、機械学習アルゴリズムにオンライン学習 (逐次学習) の一つである Soft Confidence-Weighted Learning (SCW)[8] を利用した。

本研究では、更に進んで、二重マッピングと SCW[8] による機械学習を組み合わせることで、特に売買タイミングを正確に知るための予測騰落強度の精度向上をはかる手法を提案する。第一段のマッピング (一次マッピング) では、収束性と安定性を備えた機械学習を実現するため、一次入力データに対して線形代数の概念である特異値分解に基づいたマッピングを行い、その結果に対して機械学習を行う。第二段のマッピング (二次マッピング) では、同じ教師データと一回目の学習過程で得られる方向性判断値、及び重みを利用し、株価騰落率をより正確に反映できるような騰落強度を得るためのマッピングを行い、その結果に対して再学習を行う。次の第 2 節では本研究の基本的な考え方について述べる。第 3 節では本研究独自の学習用データセットについて述べる。第 4 節では本研究で提案した手法について述べる。第 5 節では日経平均株価指数を対象にした予測実験について述べる。

2 基本的な考え方

本研究では、文献 [9] でも述べた通り、以下の 6 つの考え方に基づいた株価変動の予測を行う。

1) 予測しやすい、かつリスクが低い銘柄・指数だけを予測の対象とする。これまで、日経平均株価指数 [10] を最も安定的に学習・予測できることを実験で確認しており、今回もこの指数を主に扱う。日経平均株価指数に一对一に連動する ETF が幾つもあり、流動性が非常に高く、いつでも売買でき、よく知られている指数でもあるため、実用的な株式投資対象としても好都合である。それに比べて個別銘柄の株価はいろいろな要素の影響を受ける。次の日に上がると予測して株を購入し、次の市場が始まる前に悪材料が出た場合、予測が大きく外れることになる。格付け機関によってレーティングが変更されたり、目標株価が僅かに変更されただけで株価が激変し、関連銘柄にも影響が出る場合がよくある。特に個別銘柄に関しては市場ではなく、格付け機関によって価格が形成される場合も有りである。

2) 一つの銘柄の株価は独立に動くことは殆どなく、複数の銘柄と直接または間接的に影響しあって相関的に変動する。日経平均株価指数に関して言えば、その構成銘柄の株価変動の影響を受けやすい。本研究では、一つの銘柄の値動きを予測するためには関連性の高い複数の銘柄の時系列データの組み合わせを入力データとし、その関連銘柄群 (ベース銘柄集) を一定の手順 (後述) に基づいて常に機械的に選択し直す。株価は、需給、株価材料、投資家心理、為替、経済政策、経済状況や地政学、要人の発言、マスメディアや SNS からのニュース (例えば偽ニュースであっても) など様々な要因の影響を受けることは周知の通りであるが、第 1 節で述べたように、現在のマーケットを取り巻く環境は劇的に変化しており、株価変動要因になり得る何らかの出来事が発生した場合、瞬時に株価に反映されるため、基本的にベース銘柄集の株価時系列データだけを学習の入力データとする。

3) 学習期間 (図 1 の (c)) として予測ポイント (図 1 の (b)) から数日間離れた最終学習ポイント (図 1 の (a)) より過去 3600 日分 (約 15 年間) の株価時系列データを使用するが、ベース銘柄集と最終学習ポイントを常に更新し続けることにより、最新の株価変動要因が株価時系列データに反映されている状態を可能な限り維持し続ける。

4) 本研究では、一日の株価変動ではなく、数日間のウィンドウ内の株価変動を学習、又は予測のターゲットにする。このウィンドウをターゲットウィンドウと呼び、学習ターゲットウィンドウ (図 1 の (d)) と予測ターゲットウィンドウ (図 1 の (e)) の二つに分けて考える。日数がターゲットウィンドウのサイズになる。ここでもやはり予測しやすいターゲットウィンドウサイズを

考える。

5) 学習ポイント (図1の(a)) から見た過去一定期間内の株価のトリプルボトムとトリプルトップの学習ポイントから見た騰落率を入力データの一次属性値 (feature) とし、その期間を属性ウィンドウ (図1の(f)) と呼ぶ。そして、ベース銘柄集の各銘柄の属性値の組み合わせを一次入力データとする。

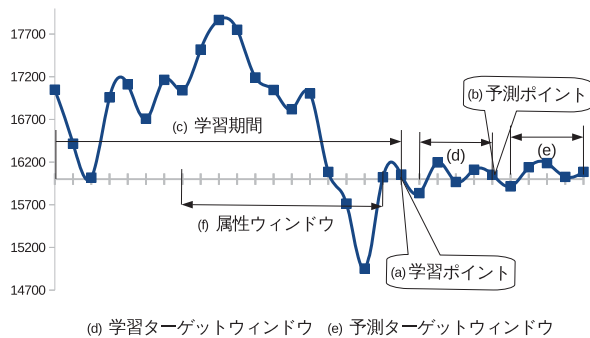


図1: 学習ポイント・予測ポイント・ウィンドウ

6) 本研究では、株価変動の方向性、即ち次の期間内に株価が上るか下がるかだけでなく、売買タイミングの割り出しにも重点をおいた予測を行う。騰落の程度を表すことのできる定量的な指標として「騰落強度」(式(2))を使用し、騰落強度が大きければ大きいほど、売買タイミングに近付いたと判断する。

SCWを含めたオンライン学習では、入力 $x_t \in \mathcal{R}^d$ と機械学習によって更新される重み $w \in \mathcal{R}^d$ に対して、

$$\hat{y}_t = \text{sgn}(x_t \cdot w), \hat{y}_t \in \{+1, -1\} \quad (1)$$

でもって入力 x_t が属するクラスを予測する。

本研究では、全ての入力 x_t が d 次元空間の点であり、それらの点が法線ベクトル w を持つ超平面によって二つに分離され、点と分離超平面との距離が大きければ大きいほどそのクラスへの属性が強いと見做し、その距離(符号付き)を「騰落強度」と定義する。即ち、騰落強度 FS (Fluctuation Strength) を次のように定義する。

$$FS = \frac{x_t \cdot w}{|w|} \quad (2)$$

ここで、 $|FS|$ は騰落の強さ、 $\text{sgn}(FS)$ は上がるか下がるかの方向性になる。しかし、騰落強度が最も大きい場合、その騰落率も一番大きくなるという訳ではなく、その傾向が強いを表す量になる。騰落強度の妥当性に関しては、先行研究 [9] での一連の実験によって検証済みであるが、本研究の主な目的の一つは株価騰落率をより忠実に反映できるような騰落強度を獲得することである。DL、あるいは最小二乗法を用いた回帰計算等によって株価騰落率そのものを推定することも考えられるが、本研究の実験範囲内では期待できる結果が得られなかった。

3 機械学習用データセットの準備

機械学習を行う場合、学習アルゴリズムと並んで学習のためのデータセットがその学習と予測結果に大きな影響を与えることがよくある。さまざまな機械学習の典型的なアルゴリズムの例題プログラムとそれ用に提供されるデータセットを Web などから見つけては実行すると、見事な結果が出たりする。さすがに今どきの機械学習であると感激したりもする。しかし、いざその学習アルゴリズムのサンプルプログラムを自分の研究中のデータセットに合わせてカスタマイズし実行してみると、期待はずれの結果が出てきたりする。従って、実用レベルの機械学習や予測結果を出すためには独自の学習データセットを用意することが重要になってくる。

本研究では次のようにして独自の学習データセットを用意する。まず第一に、ベース銘柄集を自動的に選択する。第二に、先行研究 [9] と同様、ベース銘柄集の各銘柄の属性ウィンドウ (図1の(f)) におけるトリプルボトムとトリプルトップの学習ポイント (図1の(a)) から見た騰落率の組み合わせを入力データの一次属性値とする。第三に、従来の研究では株価の騰落率に関して 0 を基準にし、学習ターゲットウィンドウ (図1の(d)) の学習ポイントから見た騰落率を計算し、騰落率 ≥ 0 は「上がり」(+1)、そうではない場合は「下がり」(-1) とラベル付けをして学習と予測を行うことが多いが、ここでは「上がり」と「下がり」の基準値を別々に決め、学習と予測も別々に行う。第四に、入力データの一次属性値に対して特異値分解の概念に基づいたデータ変換を行い、新たな入力データを生成する。これは所謂二重マッピングの第一段に相当する。

以上の第二の方法で決める一次属性値の妥当性であるが、本研究の可能な範囲内でいくつかの種類値を属性値として学習と予測実験を行い、トリプルボトムとトリプルトップベースの属性値が最もよいことが分かればよしとする。学習データセットの準備の詳細に関しては以下の各サブ節で述べることにする。

3.1 ベース銘柄集の自動選択

日経平均株価指数、あるいは特定の業種に連動するように設定されたETFなどは独立に動くことは殆どなく、複数の銘柄と直接または間接的に影響しあって相関的に変動すると思われる。日経平均株価指数を考えた場合、構成銘柄リストに含まれていない場合でさえその可能性がある。従って、一つの銘柄の値動きの予測を行う場合、連動性の高い銘柄群の時系列データの組み合わせを学習の入力データとするのが自然であろう。ここでは、そのような銘柄郡をベース銘柄集と呼ぶ。

ベース銘柄集をどの株式市場の上場銘柄から選ぶのか、国内外から同時に選ぶのかなどを検討する必要がある。

るが、今回は日経平均株価指数の変動を学習と予測の対象とし、東京証券取引所で取引されているものの中で一定の条件を満たす銘柄をベース銘柄集の選択対象とする。

ベース銘柄集を次のような手順で選択する。便宜上、選択対象の全銘柄の集合を S 、学習期間を T 、銘柄 $s \in S$ の学習期間中の売買成立日数を T_s 、 s の最大連続売買不成立日数を N_s ($T \geq T_s + N_s$)、基準最低取引成立日数を T_0 、基準最大連続不成立日数を N_0 とする。

Step0 予測対象銘柄 $s_0 \in S$ を一つ決定する。話を進めやすくするため、全期間中において売買不成立日のない銘柄を選ぶ。即ち、 $N_{s_0} = 0$ になるような銘柄を選ぶ。日経平均株価指数は正にこの条件を満たす。

Step1 $S_1 = \{s \in S \mid T_s \geq T_0, N_s \leq N_0\}$ とする。即ち、できるだけトータルで売買不成立日が少くかつ連続の売買不成立日の少ない銘柄を選ぶ。 $s \in S_1$ に売買不成立がある場合、補間を行う。

今回の研究では $T = 3600$ 、 $T_0 = 0.99T = 3564$ 、 $N_0 = 1$ としており、そうした場合、 $|S_1|$ は大抵 1100 前後におさまる。

Step2 S_1 から一様平均流動性の高い上位 $P_1\%$ の銘柄を選び、 S_2 とする。ある銘柄 $s \in S_1$ の予測ポイントから過去 T 日間の株価終値、及び出来高の時系列をそれぞれ $p_t, v_t, t = 1, 2, \dots, T$ とした場合の s の一様平均流動性 $AvgLiq(s)$ を次のように定義する。

$$AvgLiq(s) = \left(\prod_{k=0}^K \left(\frac{1}{R} \sum_{t=1}^R p_t \cdot v_t \right) \right)^{\frac{1}{K+1}}$$

ここで、 $K \geq 0, 2^{K+1} \leq T, R = \lfloor \frac{T}{2^k} \rfloor$ 。

Step3 S_2 から一様平均騰落率の高い上位 $P_2\%$ の銘柄を選び、 S_3 とする。ある銘柄 $s \in S_2$ の予測ポイントから過去 T 日間の株価終値時系列を $p_t, t = 1, 2, \dots, T, T+1$ とした場合の s の一様平均騰落率 $AvgPC(s)$ を次のように定義する。

$$AvgPC(s) = \left(\prod_{k=0}^K \left(\frac{1}{R} \sum_{t=1}^R \left| \frac{p_t - p_{t+1}}{p_{t+1}} \right| \right) \right)^{\frac{1}{K+1}}$$

ここで、 $K \geq 0, 2^{K+1} \leq T, R = \lfloor \frac{T}{2^k} \rfloor$ 。

Step4 $S_3 \cup \{s_0\}$ から s_0 の s に対する一様平均絶対ベータ値の高い上位 $P_3\%$ の銘柄を選び、それらを s_0 のベース銘柄集 $S(s_0)$ とする。前の各 **Step** で s_0 が落とされる可能性があるため、ここで s_0 がベース銘柄集に入るようにしておく。本研究では、 $|S(s_0)| = N$ とし、 $P_1 = P_2 = P_3$ 、

$P_1\% \cdot P_2\% \cdot P_3\% \cdot |S_1| = N$ になるように P_1, P_2, P_3 の値を決めている。例えば $N = 320$ の場合、 $P_1 = P_2 = P_3 \simeq 66$ 。日経平均株価指数の場合、ベース銘柄集のサイズを 320 にしているが、対象銘柄によって増減する必要がある。

s_0 とある銘柄 $s \in S(s_0)$ の予測ポイントから過去 $T + L$ 日間の株価終値時系列をそれぞれ $p_t^0, p_t^s, t = 1, 2, \dots, T, T+1, \dots, T+L$ とし、 s_0 の s に対する一様平均絶対ベータ値 $AvgBeta(s_0 | s)$ を次のように定義する。

$$AvgBeta(s_0 | s) = \left(\prod_{k=0}^K \left| \frac{\alpha_k \cdot \beta_k}{\beta_k} \right| \right)^{\frac{1}{K+1}}$$

$$\alpha_k = \left\{ \frac{p_1^0 - a_1^0}{a_1^0}, \frac{p_2^0 - a_2^0}{a_2^0}, \dots, \frac{p_R^0 - a_R^0}{a_R^0} \right\}$$

$$\beta_k = \left\{ \frac{p_1^s - a_1^s}{a_1^s}, \frac{p_2^s - a_2^s}{a_2^s}, \dots, \frac{p_R^s - a_R^s}{a_R^s} \right\}$$

$$a_t^0 = \frac{1}{L} \sum_{i=1}^L p_{t+i}^0, \quad a_t^s = \frac{1}{L} \sum_{i=1}^L p_{t+i}^s$$

ここで、 $K \geq 0, 2^{K+1} \leq T, R = \lfloor \frac{T}{2^k} \rfloor$ 、 L は属性ウィンドウのサイズ、 a_t^0 と $a_t^s, t = 1, 2, \dots, T$ はそれぞれ L 日間の移動平均値である。 $K (= 7)$ を折り畳み回数と呼ぶことにする。一般的に個別銘柄の日経平均に対するベータ値を計算するが普通であるが、ここではその逆である。

以上の手順で日経平均株価指数のベース銘柄集 $S(s_0)$ を自動的に選択した場合、 $S(s_0)$ には日経平均構成銘柄の約 30% 前後が含まれておらず、 S_1 に含まれる日経平均構成銘柄を全部入れた上で残りの銘柄を上記の手順で選択してできるベース銘柄集 $S'(s_0)$ と $S(s_0)$ とで別々に学習・予測実験を行ったところ、 $S(s_0)$ のほうが良い結果になっており、この選択手順が妥当であると言える。また、日経平均株価指数だけに限って言えば、その構成銘柄の一部が場合によって必ずしも指数の変動により大きな影響を与えるとは言えないことも意味する。

3.2 一次属性値の決定

機械学習の場合、何を入力データの構成要素にするのが重要なファクターの一つであり、その構成の仕方によっては不十分な学習になったり、過学習に陥ったり、識別率に大きな影響を及ぼしたりする。入力データの構成要素として移動平均値と RSI (Relative Strength Index) の組み合わせを用いたり [6]、 n 日間の株価時系列データをそれぞれウィンドウ内と全学習期間内で正規化したデータの組み合わせを用いた [11] 研究がある。

本研究では、ベース銘柄集の各銘柄の属性ウィンドウ (図1の(f))内のトリプルボトムとトリプルトップの騰落率を一次属性値の構成要素とし、それらの組み合わせを一次属性値として使用する。

ベース銘柄集の一銘柄のある時点(学習ポイントか予測ポイント) t_0 における株価終値を p_0 , その属性ウィンドウのサイズを $L = 3 \times 2 \times stride + slack$, 属性ウィンドウ内の株価終値時系列を $p_t, t = 1, 2, \dots, L$ とした場合のトリプルボトムの時点 t_0 に対する騰落率を次のように求める。

Step1 $c_t = \frac{(p_t - p_0)}{p_0} 100, t = 1, 2, \dots, L$ とする。

Step2 $t_1, t_2, t_3 \in \{1, 2, \dots, L\}, t', t'' \in \{t_1, t_2, t_3\}, |t' - t''| \geq stride$ を満す下位の $\{c_{t_1}, c_{t_2}, c_{t_3}\}$ を選択する。即ち、時系列的に隣ではなく、互いに $stride$ 日以上離れた下位三つの値を選択する。

Step3 $\{c_{t_1}, c_{t_2}, c_{t_3}\}$ を元の時系列順に戻し、トリプルボトムの騰落率 (l_1, l_2, l_3) を得る。

上記の手順と同様、上位3つの値を選択することでトリプルトップの騰落率 (h_1, h_2, h_3) を求めることができる。今回は $stride = 2, slack = 2 (L = 14)$ としているが、ターゲットウィンドウ (図1の(d), (e))のサイズに合わせて $stride$ と $slack$ の値を適切に調整する。

次に、予測対象銘柄 s_0 のベース銘柄集

$$\mathcal{S}(s_0) = \{s_1, s_2, \dots, s_N\}, N = |\mathcal{S}(s_0)|$$

において、 $s \in \mathcal{S}(s_0)$ の時点 t のトリプルボトムとトリプルトップの騰落率を $\alpha_t^s = (l_{t1}^s, l_{t2}^s, l_{t3}^s, h_{t1}^s, h_{t2}^s, h_{t3}^s)$ とし、 $x_t^0 = (\alpha_t^{s_1}, \alpha_t^{s_2}, \dots, \alpha_t^{s_N})$, $t = 1, 2, \dots, T$ を入力データの一次属性値とする。一次属性値 $x_t^0, t = 1, 2, \dots, T$ を行に持つ行列を X^0 とする。便宜上、この行列を一次属性値行列、 x_t^0 を一次属性値ベクトルと呼ぶことにし、以降、同様の名称で参照する。

3.3 ラベル付け基準値の決定

入力データの一次属性値が決まったところで、次に問題になるのは教師データであるラベルを何を基準に決めるかということである。第2節でも述べたように、一日の株価変動ではなく、数日間のターゲットウィンドウ内の株価変動の学習と予測を行う。また、「上がるかどうか」と「下がるかどうか」の基準値を別々に決め、学習と予測も別々に行う。

学習・予測対象銘柄 s_0 の「上がり」、「下がり」それぞれの基準値は次のように決める。

まず、学習期間内の全学習ポイント (図1の(a)) から見たターゲットウィンドウ (図1の(d)) の平均騰落率 $PC(t)$ (Percentage change) を計算する。即ち、 p_t を学

習ポイント t の株価終値、 M をターゲットウィンドウのサイズ、 $p_{t-1}, p_{t-2}, \dots, p_{t-M}$ をターゲットウィンドウ内の株価終値の時系列とし、 $t = 1, 2, \dots, T$ に関して

$$PC(t) = \frac{(a_t - p_t)}{p_t} 100, a_t = \frac{1}{M} \sum_{i=1}^M p_{(t-i)}$$

を計算し、それらを昇順に並べ替えた

$$r_1, r_2, \dots, r_{\lfloor \frac{T}{3} \rfloor}, \dots, r_{\lfloor \frac{2T}{3} \rfloor}, \dots, r_T$$

の中の $LC = r_{\lfloor \frac{T}{3} \rfloor}$ を「下がり」の基準値 (Lower Criterion), $UC = r_{\lfloor \frac{2T}{3} \rfloor}$ を「上がり」の基準値 (Upper Criterion) とする。ラベル付けの基準値 UC と LC が決定されたところで、

$$y_t^U = \begin{cases} +1 & \text{if } PC(t) \geq UC \\ -1 & \text{otherwise} \end{cases}$$

$$y_t^D = \begin{cases} +1 & \text{if } PC(t) \leq LC \\ -1 & \text{otherwise} \end{cases}$$

$$t = 1, 2, \dots, T$$

によってそれぞれ「上がるかどうか」と「下がるかどうか」のラベル付けを行う。そして、

$$Y^U = (y_1^U, y_2^U, \dots, y_T^U)^\top, Y^D = (y_1^D, y_2^D, \dots, y_T^D)^\top$$

$X = X^0$ (第3.2節) とし、 $\{X, Y^U\}$ と $\{X, Y^D\}$ でそれぞれ「上がるかどうか」、「下がるかどうか」の学習ができるようになる。ここでも便宜上、 Y^U と Y^D をラベルベクトルと呼ぶことにする。

4 二重マッピングと二重学習

第3.2節の方法で作成される一次属性値行列 X^0 と第3.3節の方法で作成されるラベルベクトル $Y = Y^U$ (あるいは Y^D) でも学習を行えるが、期待できるほどの結果が得られない。そこで、先行研究 [9] では、

$$X^0 = U \Sigma V^\top \quad (3)$$

のように特異値分解し、 $X = X^0 V$ として属性値行列を作成し、この X とラベルベクトル Y に対して SCW アルゴリズムを利用した機械学習を行った結果、僅かではあるが実用レベルの成果が出た。即ち、一次属性値ベクトル x_t^0 を直接学習の入力とせず、 $x_t = x_t^0 V$ による変換を行い、ベクトル x_t を学習の入力とした。これは一種のマッピングであり、一次マッピングと呼ぶことにする。行列 V が一次マッパーになる。

SCW の機械学習アルゴリズムでは、入力ベクトル $x_t \in \mathcal{R}^d$ と学習によって更新される重み $w \in \mathcal{R}^d$ 対

して、 $\hat{y}_t = \text{sgn}(x_t \cdot w)$, $\hat{y}_t \in \{+1, -1\}$ を見て入力 x_t が属するクラスを予測する。本研究では $x_t \cdot w$ を方向性判断値と呼ぶことにする。この方向性判断値が限り無く 0 に近い場合でも判断を行い、方向が一致すれば、重み w の更新を行わず、次の入力を待つ。しかし、実際の株式投資では想定 of 基準値以上に上がる、あるいは基準値以下に下がるという予測ができなければ、中々決断できない¹。それが故に本研究での「上がるかどうか」、「下がるかどうか」の判断基準値を 0 ではなく、第 3.3 節で述べた方法で決めている。

しかし、それでもなお上がるならよく上るかどうか、下がるならよく下がるかどうかを定量的に計ることができれば、決断の更なる手助けになる。著者は、そのために式 (2) のように騰落強度 FS を定義し、その有用性を実験によって示した [9]。

本研究では一歩進んで、学習過程においては騰落率が大きければ大きいほど騰落強度が大きくなり、予測においては騰落強度が大きければ大きいほど騰落率も大きくなるという傾向を更に強くすることを考える。そこで、まず一次マッピングの結果に対して予備の学習 (一次学習) を行い、得られた方向性判断値、重み、及び教師データの元である騰落率を利用して新たなマッパーを作成し、その二次マッピングの結果に対して同じ SCW による再学習を行うという手法を導入する。本手法の詳細は次の通りである。

一次学習を行った結果、得られた重みベクトルを $w = (w_1, w_2, \dots, w_n)$ 、方向性判断値ベクトルを $d = (d_1, d_2, \dots, d_T)$ とすると、次の式が成り立つ。

$$X^0 V w^T = d^T \quad (4)$$

しかし、 d の各成分の絶対値の大きさと騰落ベクトル $f = (f_1, f_2, \dots, f_T)$ の各成分の絶対値の大きさの方向性が必ずしも一致しない。そこで、 f のある成分の絶対値が大きければ、それに対応する d の成分の絶対値も大きくなるような成分の数ができるだけ多くなるようにしたい。そうなれば、式 (2) の騰落強度 FS の識別能力が高まると期待できる。

上記の発想から、次の式 (5) を満たすようなもう一つのマッパー行列 M を作ればよいことになる。

$$X^0 V M w^T = \frac{|d|}{|f|} f^T \quad (5)$$

ここで、 $W = w^T w$ とし、 $W = U_1 \Sigma_1 V_1^T$ のように特異値分解し、更に式 (3) から、

$$M = \Sigma^{-1} U^T \frac{|d|}{|f|} f^T w V_1 \Sigma_1^{-1} U_1^T \quad (6)$$

になり、この M を二次マッパーとする。二つのマッパーが揃ったところで、一次属性値ベクトル x_t^0 , $t =$

¹少なくとも著者はそうである

$1, 2, \dots, T$ に対して一次マッパー V と二次マッパー M によって次の式のような二重マッピングを行い、その結果であるベクトル x_t を最終入力ベクトルとして再学習を行う。

$$x_t = x_t^0 V M, \quad t = 1, 2, \dots, T \quad (7)$$

5 予測実験

第 2 節で述べたように、予測しやすい銘柄・指数だけを予測の対象とする。これまでは、日経平均株価指数 [10] とそのベース銘柄集の各銘柄、更に日経平均株価に連動する複数の ETF に関して、学習と予測実験を行った結果、日経平均株価指数が最もよかった。その善し悪しを次の三つの観点から観察した。

一つは学習のエポック回数である。学習が 100% 終るまでにかかったエポック回数が日経平均株価指数よりも少ない回数で済んだ銘柄が銘柄数全体の 6% 前後であった。日経平均株価指数のベース銘柄集を他の銘柄のベース銘柄集として学習を行っても大きな違いがなかった。

二つ目はベース銘柄集に含める銘柄の数である。日経平均も個別の銘柄もその株価は関連性の高い銘柄と影響し合って変動することは同じであるが、日経平均に影響を与える銘柄が沢山あるのに対し、個別銘柄は必ずしもそうではない。関連性の薄い銘柄をできるだけベース銘柄集に含めないようにすると、学習用入力属性値ベクトルの次元が不十分になり、安定した学習ができない。属性値ベクトルの次元を適切なサイズにしようとする、関連性の薄い、或いは関連性のない銘柄まで含まれてしまい、学習は安定的になるが予測は安定的に行えない。

三つ目は予測の正解率である。日経平均株価の予測の正解率が実験を実施した他のどの銘柄よりも高く安定的であった。日経平均株価に連動する複数の ETF と比べても平均的に 3% 前後の差があった。第 3.1 節の手順でそれらの ETF のベース銘柄を選択すると、日経平均のベース銘柄集と 1, 2 銘柄しか差がないにも関わらずである。

しかし、本研究の主な目的は二重マッピングと二重学習の有効性を示すことであり、ここでは今回提案した手法に基づいたシステムの予測正解率と騰落強度の売買タイミングにおける識別能力に関する実験結果だけ報告する。

5.1 実験に使用したデータ

今回の実験には、東京証券取引所に上場している銘柄のうち、東証 1 部、東証 2 部、JASDAQ を問わず、入手

できる株価データ全てを使用した。それには 3800 もの銘柄が含まれているが、第 3.1 節で述べたベース銘柄選択手順の **Step1** を実行するだけでその数が 1100 前後に減り、残りの各 **Step** の実行後に 320 銘柄になるように関連パラメータの値を決めており、一定の期間内の予測を行う場合、実際に使用されるのが 370 前後の銘柄である。実験には Yahoo!ファイナンス [12] のサイトから 2001 年 1 月から 2016 年 12 月までの 16 年分の時系列データをダウンロードして使用した。その時系列データには「日付」、「始値」、「高値」、「安値」、「終値」、「出来高」の他に「調整後終値」も含まれており、そのまま実験に使用した²。

5.2 予測正解率に関する実験

システムの予測正解率が何%までなら識別能力があると判断されるかは困難であるが、この実験では、一定の予測期間内の各タイミングで「上がるかどうか」(「下がるかどうか」)に関して予測を行い、「上がる(Yes)」(「下がる(Yes)」)と予測されたタイミング(その回数を A とする)の内、本当に上がった(下がった)回数(その回数を B とする)を勘定し、 $\frac{B}{A}100$ を正解率(%)とする。本研究では、「上がるかどうか」と「下がるかどうか」に関して別々に予測を行う。つまり、二つの予測器で別々に予測を行う。

今回は、ターゲットウィンドウ(図 1 の (d, e))のサイズを 3 とし、2016 年 7 月 1 日から 2016 年 12 月 30 日までの 123 日間(6ヵ月間)を予測期間、それ以前の 3600 日間を学習期間として「上がるかどうか」と「下がるかどうか」別、及び一次マッピングと二重マッピング別の実験を行った。その結果を表 1 に示す。

表 1 の結果から、一次マッピングの場合の正解率は「上がるかどうか」と「下がるかどうか」の両方とも二重マッピングの場合より 3% 前後よいことが分かり、後者が悪化したのは二回目のマッピングで属性値ベクトルにノイズが加わったと考えられる。従って、「上がるかどうか」(「下がるかどうか」)だけを予測したい場合、二重マッピングを行わないほうがよさそうである。

マッピング	「上がる?」		「下がる?」	
	一次	二重	一次	二重
予測 Yes 回数	44	41	43	44
真の Yes 回数	36	32	34	33
正解率 (%)	81.81	78.04	79.06	75.0

表 1: 予測正解率に関する実験結果

また、二重マッピングの場合の再学習にかかったエポック回数が一回目の学習に比べて平均的に倍もかかっており、この状況からも最終入力属性値ベクトルが悪化したことが分かる。

5.3 騰落強度の識別能力に関する実験

これまで、「上がるかどうか」(「下がるかどうか」)に対して「上がる」(「下がる」)と予測された場合、その「上がる」(「下がる」)程度がどれぐらいかを定量的に測るために「騰落強度」(式 (2) を参照)を使用し、騰落強度が大きければ大きいほど、それに伴うターゲットウィンドウの騰落率も大きくなる傾向があり、それが売買タイミングの見極めに有効であることを示した [9]。従って、予測騰落強度が充分大きくなった場合、その時点で売買を決断するわけである。

本研究では、一歩進んで、騰落強度の大きさに伴って騰落率も大きくなる相関性を更に強くさせるようなマッパーを導入した。そして二重マッピングを行った場合、予測騰落強度と実際の騰落率の相関性がそうでない場合に比べて強くなったのかどうかについてやはり「上がるかどうか」と「下がるかどうか」別、及び一次マッピングと二重マッピング別の実験を行った。その結果を表 2 に示す。ここでもターゲットウィンドウのサイズや予測期間等の条件は第 5.2 節と同様である。

マッピング	「上がる?」		「下がる?」	
	一次	二重	一次	二重
相関度	0.68	0.79	0.61	0.73

表 2: 騰落強度の識別能力に関する実験結果

表 2 から、二重マッピングの場合の予測騰落強度と実際の騰落率の相関度が「上がるかどうか」と「下がるかどうか」の両方とも一次マッピングの場合に比べてかなり改善されたことが分かる。

6 結論

二重マッピングと SCW [8] による機械学習を組み合わせることで、特に売買タイミングを正確に知るための予測騰落強度の精度向上を実現する手法を提案し、その有効性を示した。また、「上がるかどうか」を主眼にした学習と「下がるどうか」を主眼にした学習に分けて学習を行った。これには二つ目的があった。

一つは「上がるかどうか」(「下がるどうか」)のラベル付け基準値の近辺での予測が僅かな「上がり」(「下がり」)であっても、できるだけ実用レベルの「上がり」(「下がり」)を得るためである。学習期間内のターゲッ

²株価時系列データに「調整後終値」もあることで大掛かりな加工を施さなくても使用でき、一貫性も保たれるため、大変助かる。この場を借りて Yahoo!ファイナンスに感謝の意を表したい。

トウィンドウの学習ポイントから見た全騰落率のソート後の $\frac{1}{3}$ 範囲内が「上がり」(「下がり」)になるようにラベル付け基準値を決め、その範囲外にも敢えて一部の「上がり」(「下がり」)を残すことでこの目的を実現した。2クラス分類問題では、教師データの正例と負例のバランスが取れていない場合、予測結果が数の多いほうに傾きやすいことが一般的に知られているが、SCW の場合はそうならない [8] になっているため、本研究でも問題にならない。今回の実験結果からもその点が確認できる。

もう一つは、株式市場が「上昇気流」に乗っている場合は緩やかに上昇するが、「調整気流」になった場合は短期間で降下する傾向にあり、騰落率が 0 以上になるターゲットウィンドウの数が多いため、「上がり」(「下がり」)のラベル付け基準値を 0 にすると、「下がり」の予測がしにくくなり、投資機会が明らかに少なくなってしまう。従って、本研究のアプローチが効果的である。

更に、株式投資は、毎日その変動を予測して毎日売買を行わなくてはならないことではないため、「上がる」、「下がる」可能性が充分高くなった時点で売買を行えばよい。確信を持った時だけ売買を行い、一定以上の利益を確保する、あるいは損だけは出さないことは投資家にとって重要である。本研究では、その観点から、これまでの研究 [9] で導入した予測騰落強度の認識力の向上をはかった手法を提案し、その有効性を示した。

今回は日経平均株価指数に関して評価実験を行ったが、他に検討したい指数としてダウ平均株価指数とナスダック総合指数がある。ダウ平均株価指数は構成銘柄数が少なく、30 銘柄数であり、本研究のアプローチの有効性が高くない可能性がある。また、ナスダック総合は加重平均指数であり、構成銘柄数が逆に大過ぎて高い正解率が期待できない可能性がある。日経平均株価指数のベース銘柄数が多すぎても少なすぎても正解率が悪くなることを関連実験で確認しており、ベース銘柄数を 320 にしたのもそのためである。従って、今後の課題として、上記の 2 つの指数に関しても同様の実験を実施し、今回提案した手法の有効性を調べたい。

謝辞

研究を進めるにあたり、その方法・考え方等多方面に渡ってご指導を頂いた恩師であり、名古屋大学名誉教授の稲垣康善先生に厚く御礼を申し上げます。

参考文献

- [1] Hinton, G. , Osindero, S. and Teh, Y. W. : A fast learning algorithm for deep belief nets, *Neural Computation*, Vol. 18, No. 7, pp. 1527–1554 (2006)
- [2] Bengio, Y., Lamblin, P., Popovici, D., and Larochelle, H.: Greedy Layer-Wise Training of Deep Networks, *Advances in Neural Information Processing Systems 19 (NIPS 2006)*, accessed February 1, 2017, <http://papers.nips.cc/paper/3048-greedy-layer-wise-training-of-deep-networks.pdf> (2007)
- [3] Chao, J., Shen, F. and Zhao, J.: Forecasting exchange rate with deep belief networks, *Proceedings of IEEE International Joint Conference on Neural Networks (IJCNN)*, pp. 1259–1266(2011)
- [4] Yeh, S. , Wang, C. and Tsai, M.: Corporate Default Prediction via Deep Learning, In *The 34th International Symposium on Forecasting (ISF '14)*, (2014)
- [5] Boser, B. E., Guyon, I. M., and Vapnik, V. N.: A training algorithm for optimal margin classifiers, *Proceedings of the fifth annual workshop on Computational learning theory (COLT '92)*, pp. 144–152 (1992)
- [6] 淵井 亮, 鍾 寧: サポートベクターマシンを利用した株価の値動き予測, 電子情報通信学会技術研究報告, *KBSE2010-42*, pp. 43–48 (2011)
- [7] Shen, S., Jiang, H., and Zhang, T.: Stock market forecasting using machine learning algorithms, *CS229 (Machine Learning) at Stanford University*, accessed February 1, 2017, <http://cs229.stanford.edu/proj2012/ShenJiangZhang-StockMarketForecastingusingMachineLearningAlgorithms.pdf> (2012)
- [8] Wang, J., Zhao, P., and Hoi, S. C. H.: Exact Soft Confidence-Weighted Learning, *Proceedings of the 29th International Conference on Machine Learning (ICML 2012)*, pp. 121–128 (2012)
- [9] 福田ムフタル: SCW を用いた株価変動予測, 人工知能学会研究会資料, SIG-FIN-016-02, (2016)
- [10] 日経平均プロフィール, <http://indexes.nikkei.co.jp/nkave/>
- [11] 小牧昇平, 白山晋: Deep Belief Network を用いた日経平均株価の予測に関する研究, 人工知能学会研究会資料, SIG-FIN-012-08, (2014)
- [12] Yahoo!ファイナンス, <http://finance.yahoo.co.jp/>