

SVM の Limit order Book への応用と精度向上

Improvement of Support Vector Machine Applied to Limit Order Book

木島隼人¹ 高田英行²

Hayato Kijima¹ and Hideyuki Takada²

¹ 東邦大学大学院理学研究科 情報科学専攻 ² 東邦大学理学部情報科学科

¹Graduate School of Information Science, Toho University

²Department of Information Science, Toho University

Abstract: Market participants place their limit/market orders by taking into account both the trajectory and current status of the limit order book. This behavior is based on the policy that the shape of the limit order book is quite informative for predicting future direction of a traded asset. In this paper, we employ Support Vector Machine combined with conformally transformed Gaussian RBF kernel to forecast the mid price dynamics. Our empirical studies show that the conformal transform methods improved the precision more than 3% in average compared to the standard Gaussian RBF kernel.

1 はじめに

高頻度データの蓄積に伴い、リミット・オーダー・ブックのダイナミクスに関心が集まっており、その確率的挙動のモデル化、マーケットインパクト関数の推定、最適執行戦略など比較的新しいテーマが議論されるようになった。

機械学習を応用する先行研究には、例えば Patel, Shah, Thakkar and Kotecha [5] がニューラルネットやランダム・フォレストなどの手法を時系列データに適用し、ミッドプライスの予測を試みている。Zheng, Moulines and Abergel [9] はリミット・オーダー・ブックに対してロジスティック回帰を用いてミッド・プライスの予測を提案している。Fletcher and Shawe-Taylor [3] や Kercheval and Zhang [4] 等ではリミット・オーダー・ブックの状態とその単位時間の変化量など様々な指標を学習データとする3クラス分類サポート・ベクター・マシンを提案している。

本研究では、[3][4]と同様に、サポート・ベクター・マシンを利用し、リミット・オーダー・ブックの状態からミッドプライスの超短期の予測を試みる。とくに、リーマン幾何学の観点から Amari and Wu [1] によって提案されたカーネル関数の共形変換により、標準的なガウスカーネルと比べて平均的に 3%以上の精度向上が見込まれる点について報告する。

2 データ

我々は、大阪取引所に上場・取引されている日経

225 先物のヒストリカルデータ (2012 年 4 月~6 月)のうち、レギュラー・セッション (午前 9 時~午後 3 時 10 分) を対象に実証分析を行った。オリジナルのデータレコードは約 20 マイクロ秒毎であるが、分析の効率化を優先し本研究では 1 秒毎のデータに加工して用いた。日経 225 先物の取引には、オークション方式が採用されており、リミット・オーダー・ブックの上で主に指値注文 (リミット・オーダー) と成行注文 (マーケット・オーダー) とそれらのキャンセルによって取引が展開されている。ここでは、呼び値の刻み幅を δ (=10 円) と表記する。時刻 t において、そのときの最良買値から $k \cdot \delta$ 円だけ離れた価格に待機中の指値売注文の量を

$$Q_{sell}^{k\delta}(t) \in Z_+$$

と定め、最良売値から $k \cdot \delta$ 円だけ離れた価格に待機中の指値買注文の量を

$$Q_{buy}^{k\delta}(t) \in Z_+$$

と定める。時刻 T_i ($0 \leq i \leq M$) においてリミット・オーダー・ブックに待機している指値売買注文の状態を表す $2n$ 次元ベクトルを次のように表す。

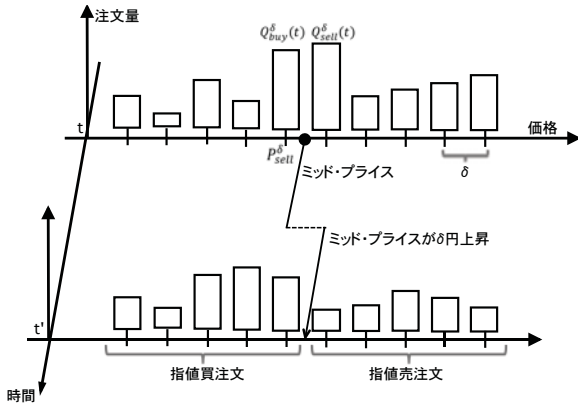
$$\mathbf{x}_i = \left(Q_{buy}^{n\delta}(T_i), Q_{buy}^{(n-1)\delta}(T_i), \dots, Q_{buy}^{\delta}(T_i), \right.$$

$$\left. Q_{sell}^{\delta}(T_i), \dots, Q_{sell}^{(n-1)\delta}(T_i), Q_{sell}^{n\delta}(T_i) \right).$$

最良買値 $P_{buy}^{\delta}(T_i)$ と最良売値 $P_{sell}^{\delta}(T_i)$ の中間値

$$\pi(T_i) := (P_{sell}^{\delta}(T_i) + P_{buy}^{\delta}(T_i))/2$$

をミッド・プライスと呼ぶ（下図参照）。



時刻 T_k 以降で初めてミッド・プライスが動いた時刻を表すインデックスを $I(T_k) \in \mathbb{Z}_+$ とする。

$$I(T_k) = \min\{j | \pi(T_{j+1}) \neq \pi(T_k), j \geq k\}.$$

3 サポート・ベクター・マシン

本研究で利用するサポート・ベクター・マシンの学習データはベクトルの列 $\{\mathbf{x}_i \in \mathbb{R}^{2n}\}_i$ であり、そのクラスを次のように付与する（教師あり学習）。

$$y_i = \begin{cases} +1 & \pi(T_{I(T_k)}) > \pi(T_{I(T_k)-1}) \\ -1 & \pi(T_{I(T_k)}) < \pi(T_{I(T_k)-1}) \end{cases}$$

すなわち、時刻 T_i 以降で初めてミッド・プライスが動いたとき、上昇していた場合は $y_i = +1$ 、下降していた場合は $y_i = -1$ と付与する。

一般に、カーネル関数 K を利用したサポート・ベクター・マシンは、入力空間 $I = \mathbb{R}^{2n}$ から特徴空間 $F = \mathbb{R}^N$ ($2n \leq N$)への写像 $\Phi: I \rightarrow F$ をうまく選ぶことによって分類能力を向上させる。学習を終えたサポート・ベクター・マシンは分離境界

$$f(\mathbf{x}) = \sum_{i \in SV} y_i \alpha_i K(\mathbf{x}, \mathbf{x}_i) + b$$

を得ており (SVはサポート・ベクター全体の集合)、新たに観測した未分類のデータ \mathbf{x} に対して

$$\mathbf{y} = h(\mathbf{x}) := \text{sign}(f(\mathbf{x})) \in \{+1, -1\}$$

により予測値を返す。

4 カーネル関数の幾何学的解釈

一般に、写像 Φ によって写されたデータ $\mathbf{z} = \Phi(\mathbf{x})$ は、特徴空間 F 内の $2n$ 次元超曲面 S 上の点とみなすことができる。 Φ が全ての方向に連続微分可能であれば、 S を $F = \mathbb{R}^N$ に埋め込まれた部分多様体と見るこ

とができ、 F から自然に誘導されるリーマン距離を持つ。リーマン距離によって、 S 上の異なる2点間の距離を測地線に沿った線積分として表すことができるが、その線素 ds は

$$(ds)^2 = \sum_{1 \leq i, j \leq 2n} g_{ij} dx^i dx^j$$

と表される。ただし g_{ij} はリーマン距離である。

$2n$ 次元の無限小ベクトル $d\mathbf{x} = (dx^1, \dots, dx^{2n})$ は、写像 Φ によって $d\mathbf{z} = \Phi(\mathbf{x} + d\mathbf{x}) - \Phi(\mathbf{x})$ に写されるので、線素は F 内で次のように表される。

$$(ds)^2 = \|d\mathbf{z}\|^2 = \|\Phi(\mathbf{x} + d\mathbf{x}) - \Phi(\mathbf{x})\|^2$$

テイラーの定理により、線素はカーネル関数 K を用いて次のように表すことができる。

$$\begin{aligned} (ds)^2 &= \sum_{k=1}^N \left\{ \sum_{j=1}^n \frac{\partial \Phi_k(\mathbf{x})}{\partial x^j} \right\}^2 \\ &= \sum_{k=1}^N \sum_{1 \leq i, j \leq 2n} \frac{\partial \Phi_k(\mathbf{x})}{\partial x^i} \frac{\partial \Phi_k(\mathbf{x})}{\partial x^j} dx^i dx^j \\ &= \sum_{1 \leq i, j \leq 2n} \left(\frac{\partial^2 K(\mathbf{x}, \mathbf{x}')}{\partial x^i \partial x'^j} \right)_{\mathbf{x}=\mathbf{x}'} dx^i dx^j \end{aligned}$$

従って、リーマン距離はカーネル関数を使って

$$g_{ij}(\mathbf{x}) = \left(\frac{\partial^2 K(\mathbf{x}, \mathbf{x}')}{\partial x^i \partial x'^j} \right)_{\mathbf{x}=\mathbf{x}'}$$

と書けることがわかる。直感的には、 $I = \mathbb{R}^{2n}$ 内の点 \mathbf{x} のまわりの局所領域は写像 $\Phi(\mathbf{x})$ によって $g_{ij}(\mathbf{x})$ 倍に伸縮されることになる。(議論の詳細は参考文献[1][6][7][8]を参照頂きたい)

5 カーネル関数の共形変換

サポート・ベクター・マシンの精度を向上させるため、Amari and Wu [1] と Wu and Amari [8] はカーネル関数を共形変換することを提案した。共形変換とは、二つのベクトルの成す角を保存するかわり、長さを変える変換のことである。Amari and Wu [1] が最初に意図した考えは、カーネル関数を共形変換することで分離境界の近くで倍率 $g_{ij}(\mathbf{x})$ を大きくし、かわりに分離境界から離れたところで倍率 $g_{ij}(\mathbf{x})$ を小さくし、全体を保ちつつ判別力を上げるというものであった。こうすることで、分離境界の近くで誤判別を減らせると期待できるからである。

具体的に、カーネルの共形変換は次で定義される。

$$\tilde{K}(\mathbf{x}, \mathbf{x}') = D(\mathbf{x})D(\mathbf{x}')K(\mathbf{x}, \mathbf{x}')$$

ただし D は I 上の正値関数で、これを適切に設定する

ことがサポート・ベクター・マシンの精度向上のために重要である。共形変換後のリーマン距離 \tilde{g}_{ij} は

$$\tilde{g}_{ij}(\mathbf{x}) = D(\mathbf{x})^2 g_{ij}(\mathbf{x}) + D'_i(\mathbf{x}) D'_j(\mathbf{x}') K(\mathbf{x}, \mathbf{x}') + 2D'_i(\mathbf{x}) D(\mathbf{x}) K'_i(\mathbf{x}, \mathbf{x}')$$

ただし、

$$K'_i(\mathbf{x}, \mathbf{x}') = \left. \frac{\partial K(\mathbf{x}, \mathbf{x}')}{\partial x_i} \right|_{\mathbf{x}=\mathbf{x}'} ; D'_i(\mathbf{x}) = \frac{\partial D(\mathbf{x})}{\partial x_i}$$

である。我々は、多くの研究が選択しているようにガウシアン RBF カーネルとその共形変換を考える。

$$K(\mathbf{x}, \mathbf{x}') = \exp\left(-\frac{\|\mathbf{x} - \mathbf{x}'\|^2}{2\sigma^2}\right).$$

更に我々は、Williams, Li, Feng and Wu [6] に従い、共形変換を定義する関数 D として

$$D(\mathbf{x}) = e^{-\kappa f(\mathbf{x})^2}$$

の形に表されるものを採用する。ここで $f(\mathbf{x})$ は既に求めてある分離境界、 κ は正の定数である。Williams, Li, Feng and Wu [6] が示しているように、このように定めた共形変換によって、分離境界の近くで $\tilde{g}_{ij}(\mathbf{x})$ を大きくし、分離境界から離れたところで小さくできるのは次のような理由である。

まず、 $g_{ij}(\mathbf{x})$ を並べた行列を $g(\mathbf{x})$ と書くとき、このリーマン距離に関する（無限小）体積要素は

$$\sqrt{\det(g(\mathbf{x}))} dx^1 \dots dx^{2n}$$

であることに注意する。従って、共形変換によって行列 $g(\mathbf{x})$ の行列式がどのように変化するかを見ればよい。ガウシアン RBF カーネルの場合は

$$g_{ij}(\mathbf{x}) = \frac{1}{\sigma^2} \delta_{ij}$$

であり、

$$\sqrt{\det(g(\mathbf{x}))} = \frac{1}{\sigma^n}$$

となる。一方、共形変換されたリーマン距離は

$$\tilde{g}_{ij}(\mathbf{x}) = \frac{D(\mathbf{x})^2}{\sigma^2} \delta_{ij} + D'_i(\mathbf{x}) D'_j(\mathbf{x}') K(\mathbf{x}, \mathbf{x}')$$

であるから、

$$\sqrt{\det(\tilde{g}(\mathbf{x}))} = \frac{D(\mathbf{x})^{2n}}{\sigma^{2n}} + \frac{D(\mathbf{x})^{2n-2}}{\sigma^{2n-2}} \sum_{i=1}^{2n} D'_i(\mathbf{x})^2$$

と書ける。定義よりベクトル $\nabla D(\mathbf{x}) = D(\mathbf{x}) \nabla \log D(\mathbf{x})$ の各成分を $D'_i(\mathbf{x})$ と書いていたので、伸縮の比率は

$$\frac{\sqrt{\det(\tilde{g}(\mathbf{x}))}}{\sqrt{\det(g(\mathbf{x}))}} = D(\mathbf{x})^{2n} \sqrt{1 + \sigma^2 \|\nabla \log D(\mathbf{x})\|^2}$$

と書き直され、 $D(\mathbf{x})$ の具体的な形から更に

$$\sqrt{\frac{\det(\tilde{g}(\mathbf{x}))}{\det(g(\mathbf{x}))}}$$

$$= \exp(-n\kappa f(\mathbf{x})^2) \sqrt{1 + 4\kappa^2 \sigma^2 f(\mathbf{x})^2 \|\nabla f(\mathbf{x})\|^2}$$

と表すことができる。方程式 $f(\mathbf{x}) = 0$ が分離境界を表すが、右辺の支配項である $\exp(-n\kappa f(\mathbf{x})^2)$ が分離境界上で最大値をとり、離れるにつれ急激に減少していくことから、当初意図した通りの性質を備えていることが確認できる (Williams, Li, Feng and Wu [6] を参照)。

6 数値計算アルゴリズム

共形変換後のカーネルを用いたサポート・ベクター・マシンの学習のためには、分離境界 $f(\mathbf{x}) = 0$ を事前に求めておく必要がある（これをプライマリー・カーネルと呼ぶ）。従って、アルゴリズムは合計2回の学習ステップを経ることになる。

高精度の予測を行うためには学習データとして十分な量を確保する必要があるが、巨大な最適化問題を数値的に解くためには多大な計算時間も必要となる。我々は、学習データとして予測時刻から遡ること過去1時間のリミット・オーダー・ブックの履歴を用いることにした。即ち、レギュラーセッションのオープニング時刻を $T_0 = 09:00:00$ 、その1時間後を $T_k = 10:00:00$ とし、区間 $[T_m, T_{m+k}]$ に記録された履歴を学習データとして使い、 $T_M = 15:10:00$ （クロージング）まで $m = 0, 1, 2, \dots$ と動かしながら予測を行う¹。ただし、実際にはミッド・プライスに変化していないと予測時点において既知の情報としてクラスを付与できないため、過去1時間の履歴が全て学習データとして使える訳ではないことに注意を要する。数値計算アルゴリズムは次のようになる。我々のデータセットでは $T_{m+1} - T_m = 1$ 秒である。

アルゴリズム

- Step 0. パラメータ C, σ, κ をセットし、学習データを区間 $[T_0, T_k]$ に得られるデータとする。
 $m = 0$ とする。
- Step 1. プライマリー・カーネル K を用いたサポート・ベクター・マシンにより、分離境界 $f(\mathbf{x})$ を求める。
- Step 2. 共形変換したカーネル \tilde{K} を用いたサポート・ベクター・マシンにより、分離境界 $\tilde{f}(\mathbf{x})$ を求める。

¹ ここで言う予測とは、「次に初めてミッド・プライスが動いたとき、上昇か下降かを予測する」ことである。

Step 3. 新しく観測されたデータ x_{m+k+1} に対して、
 $y_{m+k+1} = \tilde{h}(x_{m+k+1}) = \text{sign}(\tilde{f}(x_{m+k+1}))$ を
予測値として返す。

Step 4. $m \leftarrow m + 1$ とし時間を 1 秒進め、学習デ
ータを区間 $[T_m, T_{m+k}]$ に得られるデータと
し、 $T_{m+k} < 15:10:00$ まで Step 1 に戻る。

このアルゴリズムが実際に使えるためには、Step1
から Step 4 までの一連の計算時間が 1 秒より短く
なくてはならないが、4GHz Intel Core i7, 32GB 1867
MHz DDR3 の環境で Matlab を用いた計算では約 0.17
秒と十分な早さを保っている。

7 結果

共形変換したカーネル関数を用いたサポート・ベ
クター・マシンのパフォーマンスを評価するため、
以下で定義される指標を用いる。はじめてミッドプ
ライスが変化するとき、上昇すると予想して実際に
上昇した件数を $\sigma_{1,1}$ とする (true positive)。また、上
昇すると予想したにもかかわらず実際には下降した
件数を $\sigma_{1,-1}$ とする (false positive)。 $\sigma_{-1,1}$ と $\sigma_{-1,-1}$ も
同様に定める。このとき、指標 $\text{Precision}_{(+1)}$ は、+1
(ミッドプライスが上昇) と予想したうち、実際に
+1 であった (上昇した) 割合を表し、 $\text{Precision}_{(-1)}$ は、
-1 (下降) と予想したうち実際に -1 であった (下降
した) 割合を表す。

$$\text{Precision}_{(+1)} = \frac{\sigma_{1,1}}{\sigma_{1,1} + \sigma_{1,-1}},$$

$$\text{Precision}_{(-1)} = \frac{\sigma_{-1,-1}}{\sigma_{-1,1} + \sigma_{-1,-1}}$$

比較対象として (共形変換を施さない) ガウシアン
RBF カーネルを考える。共形変換のパラメータ
 $\kappa = 0$ がガウシアン RBF カーネルに対応している
ことを思い出しておく。 κ を大きくすればそれだけガ
ウシアン RBF カーネルからずれることになる。こ
こでは $\kappa = 1.5$ とおいたが、より適した値を探し出す
方法は今後の課題となる。

また、我々は共形変換の効果に関心があるので、
 C, σ のチューニングは行わないこととする。そこで、
 C, σ の値として $(C, \sigma) \in \{(1,1), (10,1), (100,1)\}$ の組
み合わせで数値実験を行ったあと、これら 3 つの場
合の指標 Precision の平均で評価を行う。

表 1 から表 3 は $\text{Precision}_{(+1)}$ の値を日時でプロッ
トしたものである。折れ線は (共形変換を施さない)
ガウシアン RBF カーネルを用いたサポート・ベク
ター・マシンによる結果を表し、実線は共形変換され
たガウシアン RBF カーネルを用いたサポート・ベク
ター・マシンによる結果を表す。

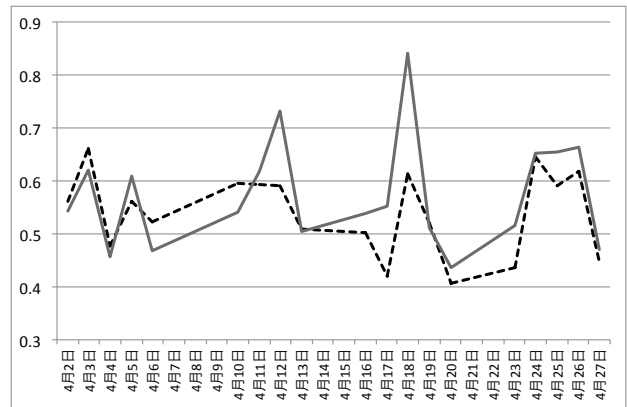


表 1 Precision₍₊₁₎ (2012年4月)

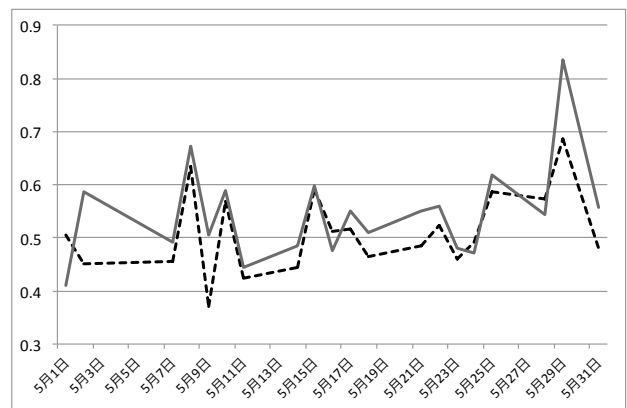


表 2 Precision₍₊₁₎ (2012年5月)

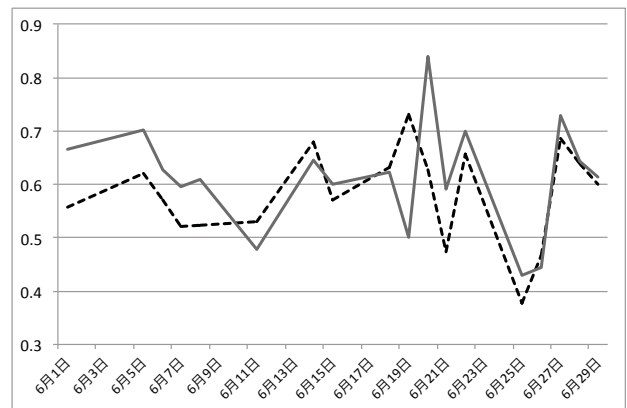


表 3 Precision₍₊₁₎ (2012年6月)

日時で両者を比較すると、常に共形変換を組み合わ
せた方法の精度が優れているわけではないが、2012
年 4 月～6 月の $\text{Precision}_{(+1)}$ を平均すると、

ガウシアン RBF カーネル … 54.340%
共形変換後 RBF カーネル … 57.757%

となり、カーネルの共形変換により Precision が 3% 以上改善されたことがわかる。

以下の表 4 から表 6 は Precision₍₋₁₎ の値を日時でプロットしたものである。

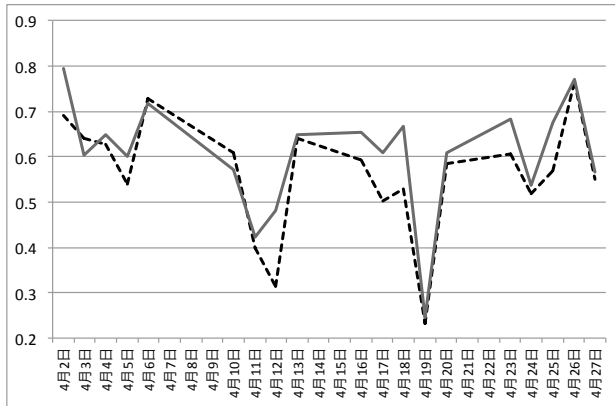


表 4 Precision₍₋₁₎ (2012年4月)

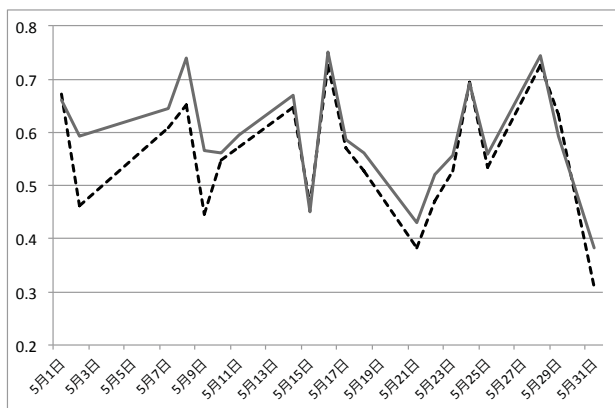


表 5 Precision₍₋₁₎ (2012年5月)

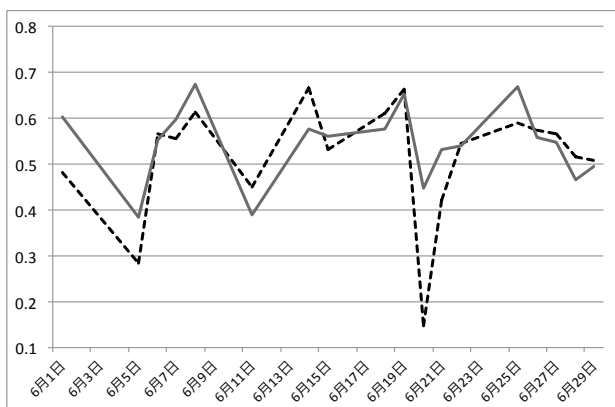


表 6 Precision₍₋₁₎ (2012年6月)

2012年4月～6月の Precision₍₋₁₎ を平均すると、

ガウシアン RBF カーネル … 54.566%
共形変換後 RBF カーネル … 58.220%

となり、こちらもカーネルの共形変換により Precision が 3%以上改善されたことがわかる。

今後は、共形変換を決定するパラメータ κ の調整により、より精度を向上させることができないか調査することになる。

参考文献

- [1] Shun-ichi Amari and Si Wu: *Improving support vector machine classifiers by modifying kernel functions*, Neural Networks, Vol. 12, 783-789, (1999)
- [2] Shangkun Deng, Akito Sakurai and Kei Shioda: *Prediction of Foreign Exchange Market States with Support Vector Machine*, 10th International Conference on Machine Learning and Applications and Workshops, Vol. 1, 327-332, (2011)
- [3] Tristan Fletcher and John Shawe-Taylor: *Multiple Kernel Learning with Fisher Kernels for High Frequency Currency Prediction*, Computational Economics, Vol. 42, Issue 2, 217-240, (2013)
- [4] Alec N. Kercheval and Yuan Zhang: *Modelling high-frequency limit order book dynamics with support vector machines*, Quantitative Finance, Vol. 15, Issue 8, 1315-1329, (2015)
- [5] Jigar Patel, Sahil Shah, Priyank Thakkar and K Kotecha: *Predicting stock and stock price index movement using Trend Deterministic Data Preparation and machine learning techniques*, Expert Systems with Applications, Vol 42, 259-268, (2015)
- [6] Peter Williams, Sheng Li, Jianfeng Feng and Si Wu: *Scaling the kernel function to improve performance of the Support Vector Machine*, Advances in Neural Networks, Volume 3496 of the series Lecture Notes in Computer Science, 831-836, (2005)
- [7] Gang Wu and Edward Y. Chang: *Adaptive feature-space conformal transformation for imbalanced-data learning*, Proceedings of the Twentieth International Conference on Machine Learning (ICML 2003), Washington DC.
- [8] Si Wu and Shun-ichi Amari: *Conformal transformation of kernel functions: A data-dependent way to improve the performance of support vector machine classifiers*, Neural Processing Letter, Vol. 15, 59-67, (2002)
- [9] Ban Zheng, Eric Moulines and Frederic Abergel: *Price Jump Prediction in a Limit Order Book*, Journal of Mathematical Finance, Vol. 3, 242-255, (2013)