オンライン予測の手法を用いた意思決定の理論と応用に関する研究

瀧本英二

東北大学大学院情報科学研究科

概 要

学習の一手法として開発され発展してきたオンライン予測の手法は，単に学習の問題にとどまらず，様々な意思決定問題に適用可能である．本研究は，この普遍性に基づき，オンライン最適化問題の広いクラスに対して，アルゴリズムの設計と解析に対する1つの方法論を与えるものである．

1 はじめに

過去の経験から将来の出来事を予測する問題は，天気予報や株価の変動予測など文字通りの予測の問題を始め，確率的言語モデルの構築，スループットの高いネットワークルーティングの設計，オークションにおける妥当な価格の決定，ゲームにおける最適混合戦略の導出など，様々な領域における意思決定問題に共通する普遍的なものである．本研究では，これらの問題におけるアルゴリズムの設計と解析を，オンライン予測モデルという枠組みで統一的に論じる．このモデルは，次のように特徴づけることができる．

オンライン性 アルゴリズムの予測と環境からのデータの提示が交互に繰り返される．

ユニバーサル性 データの生成のされ方について，確率的なものも含めて何の仮定もおかない．

最悪値評価 アルゴリズムの性能を最悪の場合で評価する．

相対評価 アルゴリズムの性能を，「最適な」予測戦略の性能と比較して相対的に評価する．

ユニバーサル性と最悪値評価の2つの性質は，オンライン予測モデルの適用範囲を極めて広いものとする一方，予測の値に意味のある精度保証を求めることを不可能にしてしまう．従って，4つめの性質は必然的なものである．すなわち，ある予測戦略のクラスが与えられていると仮定し，その中で，これまで観測したデータ系列に対して最も成績の良かった戦略の予測精度に匹敵する予測を行うことを求めるのである．ここで，個々の予測戦略は，広い意味でエキスパートと呼ばれる．

一方，理論的に導出された最悪の性能値は，実際の性能を過小評価した非現実的なものと考えられがちである．しかし，Cover のユニバーサル・ポートフォリオ理論に基づく株式投資アルゴリズムのように，最悪の性能保証という理念で設計されたアルゴリズムが，現実のデータに対しても従来の手法を凌駕する性能を示すことが十分期待できるのである [6, 16]．

これまで，複数のエキスパートが与える予測を統合し，最適なエキスパートに匹敵する予測値系列を出力するさまざまな手法が開発されている [4, 13, 10, 22]．これらの手法は，エキスパートの予測の重みつき平均に基づいて自らの予測値を求めるという共通の性質を持つ．しかし，多くの自然な問題では，統合すべきエキスパートの数が指数的となるため，この手法を直接適用することは現実的ではない．

本研究では，重みベクトルを圧縮して低次元ベクトルとして保持し，これを展開することなく予測の統合や重みの更新を効率よく模倣する手法について検討を行い，グラフの確率フローに基づく重みベクトルの次元圧縮手法を与えた [21]．これは，CTW と呼ばれる情報圧縮法で用いられる手法を著しく拡張したものとなっている．本手法は，決定木や決定グラフの

学習における過適合の現象を防ぐ「枝刈り」[17, 20]や，ネットワーク上のルーティング問題 [2, 14] に応用することができる．

一方，エキスパート数が非加算無限の場合についても適用可能なアルゴリズムの新しい設計手法を与えた．これは，ゲーム理論におけるミニマックス戦略に基づくもので，ほぼ任意のオンライン最適化問題に適用可能である．本手法を，指数分布族の定める言語モデルのオンライン推定問題に適用したところ，従来手法より最悪の性能保証が改善される場合があることを示した [18, 19]．よって，本手法をさまざまな情報圧縮法に導入することによって，その性能を改善できる可能性がある．また，この手法を HMM 推定問題に適用し，話者適応問題について一定の成果を得た [15]．

さらに，オンライン予測の新しい解析手法やモデルの拡張についても検討した．特に，投資の対象となるオプションの損失の範囲（リスク情報）が与えられている場合について，従来手法を改良し，より厳密な性能評価を与えた [7]．

本稿では，オンライン予測の一般的な枠組みとエキスパートの予測の重みつき平均に基づく統合スキームの原理について述べた後，本研究で得られた成果の一部を紹介する．

2 二値予測問題

オンライン予測の形式的な枠組みを与える前に，この枠組みで議論できる意思決定問題の典型的な例として，複数のエキスパートの予測に基づく二値予測問題について述べる．

あなたは，東京ドームで行われているあるクイズ大会の一次予選に参加している．第一問は，「自由の女神が建てられた時，女神はニューヨークで一番高い建造物だった．○か×か？」であった．答えが○と思ったら外野側，×と思ったら内野側の陣地に移動しなければならない．このように，一次予選では○×形式のクイズがたくさん出され，成績の良い上位100名が二次予選に進める．しかし，まったく見当がつかない[1]．困ったあなたは，かつてこの大会で優勝したことのあるクイズ王（エキスパート）が8人参加しているのを見つけ，彼らの動向を見てから自分の答えを決めることにした．8人のうち1人は全問正解すると仮定して，あなたの取るべき戦略は？

これは典型的な 0-1 系列のオンライン予測の問題であり，「これまで一度も間違えていないエキスパートたちの予測の多数決に従う」という戦略（二分アルゴリズムと呼ばれる）が最適となる．この戦略に従うとすると，あなたが間違える度に，全問正解のエキスパートの数は半分以下に減ることになるので，問題が何題出されようと，あなたは全部で高々3回しか間違えることはない．同様にして，一般にエキスパート数が N のとき，二分アルゴリズムの誤り回数は $\log_2 N$ 以下となることが容易に分かる．

では，全問正解のエキスパートが存在するという仮定をはずし，エキスパートの予測に何の仮定もおかないとしよう（**ユニバーサル性**）．この場合，結果の系列がエキスパートの予測に束縛されないので，いかなる予測戦略を用いても，あなたを全問不正解にする極端な例を考えることができる（**最悪値評価**）．すなわち，アルゴリズムの性能評価の基準として誤り回数の絶対値を採用すると，この問題は自明でつまらないものになってしまう．しかし，ここで求められているのは，最も成績の良いエキスパートに準ずる成績を収めることであろう（**相対評価**）．エキスパートの成績がいずれも悪ければ諦めるしかないが，1人でも成績の良いものがいた場合は，あなたの成績も良くあって欲しい．

この目標を達成する戦略として，二分アルゴリズムを拡張した重みつき多数決 [13] に基づく手法が知られており，その誤り回数の上界は $k + 8\sqrt{k \ln N} + 5 \log_2 N$ で与えられる．ここで，k は，最も成績の良いエキスパートの誤り回数である [4, 22]．この上界は，重みつき多数決アルゴリズムがたとえ全問不正解という場合でも成立する．つまり，いかなる場合でも，あなたと最適なエキスパートとの成績の差が小さく抑えられることが保証されるのである．

[1] ちなみに答えは○だそうです．

3 モデルと統合スキーム

3.1 オンライン予測モデル

2 節で述べた予測の問題は，毎時刻予測値を出力する N 個のエキスパート，それらを統合して自分の予測値を出力するアルゴリズム（学習者），および真の結果を提示する環境の間で行われる繰り返しゲームと捉えることができる．これを一般化し，オンライン予測の問題を次のように形式的に表す．

Y を真の結果値の集合，$\hat{Y}$ を予測値の集合とし，毎時刻の予測の良さを評価するための損失関数 $L : Y \times \hat{Y} \to [0,\infty]$ が与えられているとする．学習者の立場から見た，各時刻 $t = 1, 2, \ldots$ における一連の動作は次の通りである．

1. 各エキスパート $i \in \{1, \ldots, N\}$ から予測値 $x_{t,i} \in \hat{Y}$ を受け取る．
2. それらを統合して，自分の予測値 $\hat{y}_t \in \hat{Y}$ を出力する．
3. 環境から真の結果 $y_t \in Y$ を受け取る．
4. このとき，学習者と各エキスパート i は，それぞれ $L(y_t, \hat{y}_t)$，$L(y_t, x_{t,i})$ の損失を被る．

以上のプロセスが時刻 T まで行われたとすると，学習者 A の累積損失は

$$L_A^T = \sum_{t=1}^{T} L(y_t, \hat{y}_t),$$

各エキスパート i の累積損失は

$$L_i^T = \sum_{t=1}^{T} L(y_t, x_{t,i})$$

でそれぞれ与えられる．学習者の目標は，最適なエキスパートとの損失との差，すなわち相対損失

$$R_A^T = L_A^T - \min_{1 \le i \le N} L_i^T$$

をなるべく小さくすることである．

例えば，2 節で述べた予測の問題は，$Y = \hat{Y} = \{0,1\}$，$L(y, \hat{y}) = |y - \hat{y}|$ という予測ゲームに相当する．

3.2 統合スキーム

Vovk は，Aggregating アルゴリズム（以下，AA）と呼ばれる学習者の統合スキームを提案し，3.1 で述べた任意のゲームに対し，ある緩い仮定のもとで最適な予測を行うことを示した [22]．しかし，AA は複雑なので，ここでは，簡易版の重みつき平均アルゴリズム（以下，WAA）を紹介する [11]．WAA の性能は AA よりも若干劣るが，単純で，その動作が損失関数に依存しないことや，予測値がエキスパートの予測の線形結合で表されるなど，カーネルによる拡張手法との整合性が良いため，実際の応用では良く用いられている．

WAA は，時刻 t において，各エキスパート i に重み $v_{t,i}$ を次のように割り当てる．

$$v_{t,i} = \frac{v_{1,i} e^{-\eta L_i^{t-1}}}{\sum_{j=1}^{N} v_{1,j} e^{-\eta L_j^{t-1}}} \tag{1}$$

ここで，$L_i^{t-1} = \sum_{q=1}^{t-1} L(y_q, x_{q,i})$ はエキスパート i のそれまでの累積損失，$\eta \ge 0$ は学習定数と呼ばれるパラメータである．(1) の分母は，重みの和が 1 となるための規格化定数である．損失の大きいエキスパートほど，小さい重みが割り当てられることに注意されたい．初期重み $v_{1,i}$ は，エキスパート i の信頼度に応じて任意に選んでよいが，通常は一様な重み（$1/N$）が選ばれる．さて，エキスパートの予測 $\boldsymbol{x}_t = (x_{t,1}, \ldots, x_{t,N})$ が与えられたとき，WAA は，重みつき平均

$$\hat{y}_t = \sum_{i=1}^{N} v_{t,i} x_{t,i} \tag{2}$$

を自らの予測値として出力する．

$Y = \hat{Y} = [0, 1]$ で，損失関数 L が凸の場合に，WAA の性能は良く解析されており，学習定数 η を適当に定めることにより，その相対損失は，ゲームに依存して決まる値 c を用いて

$$R_{\mathrm{WAA}}^T \le c \ln N \tag{3}$$

で与えられる [11]．

4 構造的エキスパートの統合

エキスパートが何らかの構造から定まり，互いに強い相関がある場合，その数 N が指数的に大きくて

も，重みを圧縮して保持することによって，AA や WAA を効率よく模倣できる可能性がある．しかも，(3) より，N が損失に寄与する大きさは $O(\ln N)$ 程度なので，損失はそれほど大きくならない．本研究では，重みベクトルの圧縮手法の 1 つとして，グラフの確率フローに基づく手法を開発した．その原理の概略と応用を以下に示す．

4.1 道カーネル法

さまざまな意思決定問題は，グラフ上の道を選択する問題として表される．これを次のようにモデル化する．始点 s と終点 t が指定されたグラフ G が与えられているとし，何らかの方法で，各 s-t 道 P に $\hat{Y}$ の予測値 $x_{t,P}$ が対応していると仮定する．ここで，道 P の予測値は時刻 t に依存してよいとする．目標は，最適な道 P の成績に匹敵する予測を行うことである．

この問題に対する自明な解決方法は，各 s-t 道をエキスパートとみなして統合アルゴリズム AA や WAA を適用することである．しかし，グラフの道の数は一般に指数的であるため，直接，重み $v_{t,P}$ を保持し，重みつき平均を計算することは非効率である．

そこで，各辺 e に遷移確率（確率フロー）$a_{t,e}$ を割り当てることによって，各道 P の重みを

$$v_{t,P} = \prod_{e \in P} a_{t,e}$$

のように間接的に表す．これは，始点 s から遷移確率 a_t に従って終点 t に達するまでランダムウォークを行ったとき，道 P をたどる確率に等しい．まず，2 つの辺空間上のベクトル a_t と a'_t が定める道空間上のベクトル v_t と v'_t の内積（道カーネル）

$$K(a_t, a'_t) = v_t \cdot v'_t = \sum_P \prod_{e \in P} a_{t,e} a'_{t,e}$$

を計算する効率の良い手法を与えた．そして，道（エキスパート）の予測値の損失が加算的であるとき，すなわち，辺空間上のベクトル ℓ_t が存在して，任意の P に対する損失が

$$L(y_t, x_{t,P}) = \sum_{e \in P} \ell_{t,e}$$

のように表せるとき，(1) で与えられる関係が成立するような遷移確率 a_t の更新が道カーネルを用いて表せることを示した．従って，(2) で与えられる予測値の計算を効率よく行うことができれば，道空間上の重みを保持することなく，WAA を効率よく模倣することができる．

4.2 決定木の枝刈り

機械学習において，最も多く用いられる仮説表現の一つが決定木である．決定木を学習するとき，まず大きな木を生成し，その後，過適合を抑えるため，適当な枝刈りを行い，木のサイズを小さくすることがよく行われる．一般に，枝刈りを行うことによって，決定木の汎化能力が増すことが知られている．ここでは，最適な枝刈り木に匹敵する予測を行うオンライン予測の問題について考える．

決定木 G が与えられているとし，これを固定する．各時刻 t において，G の各頂点には何らかの分類規則が，各辺には予測値 $z_{t,e}$ がそれぞれ割り当てられてるとする．G にインスタンスが与えられると，各頂点に割り当てられている分類規則によって，根から葉に至る道 P_t が指定される．G の枝刈り R とは，G の辺の集合で，G の任意の道 P とちょうど 1 つの辺を共有するものである．すなわち，$|P \cap R| = 1$．時刻 t におけるインスタンス P_t に対する枝刈り R の予測値 $x_{t,R}$ を，P_t と R の共有辺 e（$P_t \cap R = \{e\}$）を用いて，$x_{t,R} = z_{t,e}$ と定義する．目標は，$x_{t,R}$ を統合して，最適な枝刈り木の総損失との差がなるべく小さくなるように予測 $\hat{y}_t$ を行うことである．明らかに，各枝刈り R をそれぞれエキスパートとみなして統合アルゴリズム WAA を適用すれば，(3) で示した精度保証を持つ予測を行うことができる．しかし，一般に枝刈りの数は指数的であるため，直接 WAA を適用することは非効率である．

この問題に対し，G の双対グラフ G^D を考える．双対グラフ G^D は始点 s と終点 t を持ち，G^D の s-t 道と G の枝刈りが一対一対応し，G^D の枝刈りと G の道が一対一対応することが知られている．すると，エキスパート R は，G^D では s-t 道に対応し，その損失はインスタンス P_t との共有辺 e を用いて $L(y_t, x_{t,R}) = L(y_t, z_{t,e})$ と表されるので，加算的であ

ることが分かる．また，辺空間上のベクトル z'_t を，$e \in P_t$ のとき $z'_{t,e} = z_{t,e}$，$e \notin P_t$ のとき $z'_{t,e} = 1$ と定義すると，$x_{t,R} = \prod_{e \in R} z'_{t,e}$ と表せるため，枝刈りの予測の重みつき平均は，

$$\hat{y}_t = \sum_R v_{t,R} x_{t,R} = K(a_t, z'_t)$$

のように道カーネルの値そのものとなる．従って，道カーネル手法を用いて，WAA を効率よく模倣することができる．

また，この手法は，決定グラフ G が平面グラフの場合に拡張することができる．

4.3 情報圧縮と CTW 符号化法

アルファベット $\Sigma = \{1, \ldots, K\}$ 上の確率的言語モデルのクラス $\Theta = \{\theta_1, \ldots, \theta_N\}$ が与えられているとする．各モデル θ_i は，各 $t \geq 1$ に対し，長さ t の系列 Σ^t 上の確率分布を指定するものである．特に，記号列 $(y_1, \ldots, y_{t-1}) \in \Sigma^*$ が与えられたとき，次の記号 y_t が j となる条件付確率を $x_{t,i}(j)$ で表す．すなわち，

$$x_{t,i}(j) = \Pr(y_t = j \mid y_1, \ldots, y_{t-1}, \theta_i).$$

従って，モデル θ_i が系列 $S = (y_1, \ldots, y_T) \in \Sigma^*$ に割り当てる確率は，

$$\Pr(S|\theta_i) = \prod_{t=1}^{T} \Pr(y_t \mid y_1, \ldots, y_{t-1}, \theta_i) = \prod_{t=1}^{T} x_{t,i}(y_t)$$

で与えられる．例えば，N グラムモデル，隠れマルコフモデル，確率文脈自由文法などが確率的言語モデルの例として挙げられる．

確率的言語モデルは，算術符号と組み合わせるこのによって情報圧縮に用いることができる．各 $t = 1, 2, \ldots$ において，算術符号器はモデル θ_i が生成する確率分布 $x_{t,i}$ に基づいて，実際の記号 y_t に対する 2 進符号列 $\zeta_t \in \{0,1\}^*$ を出力する．時刻 T に達し，系列 S をすべて読み終えたとき，得られた符号列の連接 $\zeta_1 \cdots \zeta_T$ は S の瞬時復号可能な符号となっており，その符号長は，

$$\lceil -\log_2 \Pr(S|\theta_i) \rceil = \left\lceil \sum_{t=1}^{T} -\log_2 x_{t,i}(y_t) \right\rceil$$

で与えられる．よって，尤度の高いモデルを用いるほど，長さの短い符号が生成できることになる

そこで，Θ のモデルを統合して，最適なモデルに匹敵する性能を有する新しい言語モデルを構築する問題について考える．各時刻 $t = 1, \ldots, T$ において，学習者は各モデルの与える分布 $x_{i,t}$ を統合して，分布 $\hat{y}_t = (\hat{y}_t(1), \ldots, \hat{y}_t(K))$ を生成する．その後，実際の記号 y_t が与えられる．学習者の目標は，符号長 $\sum_{t=1}^{T} -\ln \hat{y}_t(y_t)$ が最適なモデルの与える符号長 $\min_i \sum_{t=1}^{T} -\ln x_{i,t}(y_t)$ に比べてそれほど大きくならないようにすることである．

この問題は，$Y = \Sigma$，$\hat{Y}$ を K 次元確率ベクトルの集合，損失関数を対数損失 $L(y, \hat{y}) = -\ln \hat{y}(y)$ とした場合の予測ゲームに相当する．対数損失関数は凸なので，統合アルゴリズムとして WAA を用いると，(3) で与えられる相対損失の上界が得られる．特に，このゲームでは $\eta = 1$ のとき $c = 1$ が成立し，

$$R^T_{\mathrm{WAA}} \leq \ln N$$

となる．これは，WAA の与える符号長が，最適なモデルの与える符号長に比べて，高々定数 $(\log_2 N + 1)$ ビットしか大きくならないことを意味する．

さて，ある N グラムモデルを 1 つ固定する．N グラムモデルは，直前の $N-1$ 記号列をインスタンスとする決定木で，各辺に，アルファベット Σ 上の確率分布が割り当てられている．従って，前節の手法を用いることにより，最適な枝刈りモデルの与える符号長に匹敵する符号化を行うことができる．これは，CTW 符号化法で用いられている手法と本質的に同じである．

4.4 動的ルーティング

次のような動的ルーティング問題について考える．始点 s と終点 t が指定された有向グラフ G が与えられている．各時刻 $t = 1, 2, \ldots$ において，学習者は G の各辺 e に遷移確率 $a_{t,e} \in [0,1]$ を割り当てる．このとき，s から t に至る道 P に沿ってパケットが運ばれる確率は，$v_{t,P} = \prod_{e \in P} a_{t,e}$ で与えられる．その後，各辺 e に対する遅延時間 $d_{t,e} \in [0,1]$ が明らかとなる．道 P の遅延時間は $y_{t,P} = \sum_{e \in P} d_{t,e}$ で与えら

れ，時刻 t におけるパケットの転送時間の期待値は

$$\sum_P v_{t,P} y_{t,P}$$

となる．学習者の目標は，時刻 T を終えたときの総転送時間の期待値が，最適な道（静的ルーティング）の総転送時間 $\min_P \sum_{t=1}^T y_{t,P}$ に比べてそれほど大きくならないようにすることである．

各道 P にエキスパートを対応させ，そのエキスパートは常に確率 1 で道 P を選ぶものと考えると，この問題は，$Y = [0, \max_P |P|]^N$，$\hat{Y}$ を道上の確率分布，損失関数を内積損失 $L(y, \hat{y}) = y \cdot \hat{y}$ とした場合の予測ゲームとみなすことができる．ここに，N は道の総数である．

エキスパート P の予測ベクトル $x_{t,P}(P')$ は，$P' = P$ のとき 1，$P' \neq P$ のとき 0 と表せるので，その損失は $L(y_t, x_{t,P}) = y_{t,P} = \sum_{e \in P} d_{t,e}$ となり加算的であることが分かる．また，アルゴリズムの予測は $\hat{y}_t = \sum_P v_{t,P} x_{t,P} = v_t$ であるので，カーネル手法を用いて，WAA を効率よく模倣することができる．内積損失は凸関数ではないので，WAA の相対損失は (3) の形にはならないが，学習定数 η を適当に定めることによって，

$$R^T_{\mathrm{WAA}} \le \sqrt{2BL^* \ln N} + B \ln N$$

が得られる．ただし，$B = \max_P |P|$，$L^* = \min_P \sum_t y_{t,P}$ である．

5 おわりに

本稿では，オンライン予測の原理といくつかの応用例を概観した．現在，凸最適化理論を用いたアルゴリズムの設計と解析手法 [5] やゲーム理論を用いた最適者追随法 [9] などが提案され，オンライン予測の原理に対する理解がさらに深まっている．また，モデルの拡張と一般化もすすんでいる [8, 1]．一方，構造的エキスパートの統合の可能性については，まだ不明な部分が多い．例えば，決定リストのクラスについて，WINNOW 等 [12, 3] のように，効率よく AA に匹敵する損失を持つ手法が存在するかどうかは重要な未解決問題となっている．また，スキーレンタル問題や k サーバ問題に代表される，競合比に基づくオンライン最適化問題の分野との関連についてもよく分かっていない．この部分が明らかになれば，本手法が意思決定問題に対するより強力な方法論を与えることは間違いない．

参考文献

[1] P. Auer, N. Cesa-Bianchi, Y. Freund, and R. E. Schapire. The nonstochastic multiarmed bandit problem. *SIAM Journal on Computing*, 32(1):48–77, 2002.

[2] B. Awerbuch and R. Kleinberg. Adaptive routing with end-to-end feedback: Distributed learning and geometric approaches. In *36th STOC*, 45–53, 2004.

[3] T. Bylander. The binary exponentiated gradient algorithm for learning linear functions. In *10th COLT*, 184–192, 1997.

[4] N. Cesa-Bianchi, Y. Freund, D. Haussler, D. P. Helmbold, R. E. Schapire, and M. K. Warmuth. How to use expert advice. *J. ACM*, 44(3):427–485, 1997.

[5] N. Cesa-Bianchi and G. Lugosi. Potential-based algorithms in on-line prediction and game theory. *Machine Learning*, 51(3):239–261, 2003.

[6] T. M. Cover. Universal portfolios. *Mathematical Finance*, 1(1):1–29, 1991.

[7] S. Harada, E. Takimoto, and A. Maruoka. Online allocation with risk information. In *LNAI*, 3734, 343–355, 2005.

[8] M. Herbster and M. Warmuth. Tracking the best expert. *Machine Learning*, 32(2):151–178, 1998.

[9] A. Kalai and S. Vempala. Efficient algorithms for online decision problems. In *16th COLT and 7th Kernel Workshop*, *LNAI*, 2777, 26–40. Springer, 2003.

[10] J. Kivinen and M. K. Warmuth. Additive versus exponentiated gradient updates for linear prediction. *Information and Computation*, 132(1):1–64, 1997.

[11] J. Kivinen and M. K. Warmuth. Averaging expert predictions. In *4th EuroCOLT, LNAI*, 1572, 153–167, 1999

[12] N. Littlestone. Learning quickly when irrelevant attributes abound: A new linear-threshold algorithm. *Machine Learning*, 2(4):285–318, 1988.

[13] N. Littlestone and M. K. Warmuth. The weighted majority algorithm. *Inform. Comput.*, 108(2):212–261, 1994

[14] H. B. McMahan and A. Blum. Online geometric optimization in the bandit setting against an adaptive adversary. In *17th COLT, LNAI*, 3120, 109–123, 2004.

[15] J. Mizuno, T. Watanabe, K. Ueki, K. Amano, E. Takimoto, and A. Maruoka. Online estimation of hidden markov model parameters. In *LNAI*, 1967, 155–169, 2000.

[16] G. Stoltz and G. Lugosi. Internal regret in online portfolio selection. In *16th COLT and 7th Kernel Workshop, LNAI*, 2777, 403–417, 2003.

[17] E. Takimoto, A. Maruoka, and V. Vovk. Predicting nearly as well as the best pruning of a decision tree through dynamic programming scheme. *Theoret. Comput. Sci.*, 261(1):179–209, 2001.

[18] E. Takimoto and M. Warmuth. The minimax strategy for Gaussian density estimation. In *13th COLT*, 100–106, 2000.

[19] E. Takimoto and M. K. Warmuth. The last-step minimax algorithm. In *11th ALT, LNAI*, 1968, 279–290, 2000.

[20] E. Takimoto and M. K. Warmuth. Predicting nearly as well as the best pruning of a planar decision graph. *Theoretical Computer Science*, 288(2):217–235, 2002.

[21] E. Takimoto and M. K. Warmuth. Path kernels and multiplicative updates. *Journal of Machine Learning Research*, 4:773–818, 2003.

[22] V. Vovk. A game of prediction with expert advice. *J. of Comput. Syst. Sci.*, 56(2):153–173, 1998.