「技巧ミニ」アピール文書

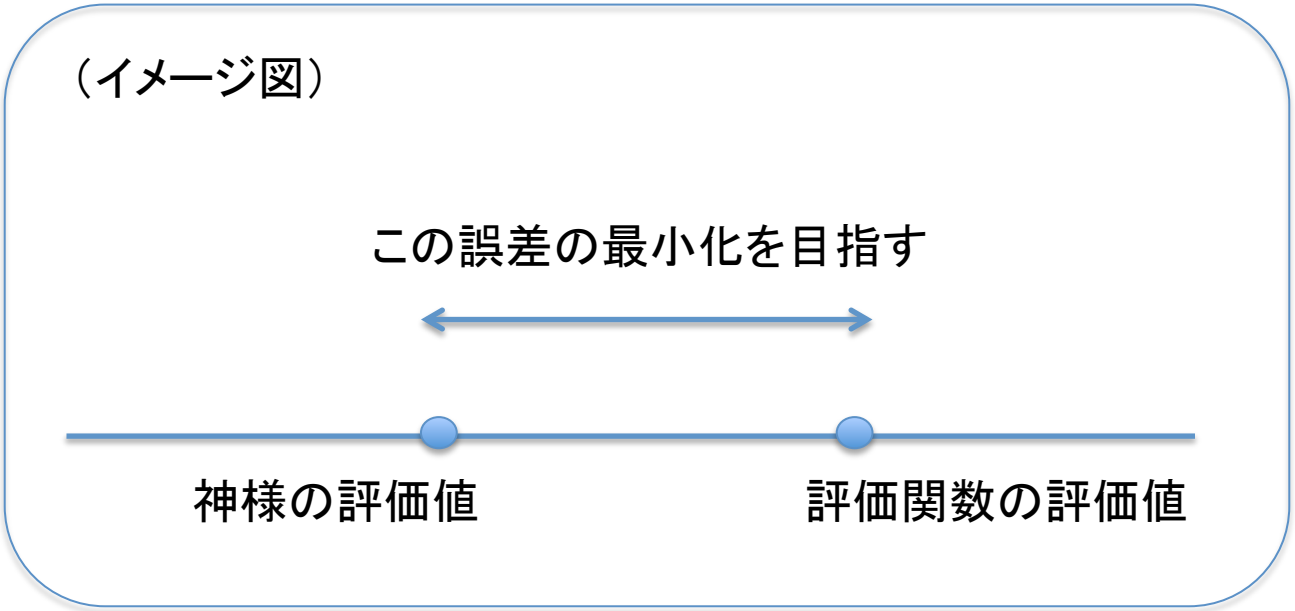
東京大学大学院法学政治学研究科
出村 洋介

「技巧ミニ」の特徴

- 新開発の「Bishop Learning」により、評価関数を学習！
  - 将棋の棋譜を用いず、
  - 自己対戦も行わないで、
  - 大規模な評価関数（パラメータ約200万）の学習に成功しました！
  - 学習時間は家庭用PCで数時間程度です

「Bishop Learning」とは？

- 「神様の評価値」との誤差最小化を目指した、半教師付き（semi-supervised）学習です
  - 神様の評価値から学ぶので、Bishop（聖職者）Learningという名前にしました



Bishop Learningの新規性

- 「半教師付き学習」という新しい分野を開拓！
  - Bonanza Methodは、教師あり学習
  - TD(λ)は、教師なし学習
- 棋譜がなくてもちゃんと学習できる！
  - Bonanza Methodではプロの棋譜が大量に必要
- 対戦を行わないので対戦相手に困らない！
  - TD(λ)では数多くの対戦が必要

既存の学習手法との比較

	Bishop Learning	Bonanza Method	TD(λ)
カテゴリ	半教師付き学習 （semi-supervised）	教師あり学習 (supervised)	教師なし学習 (unsupervised)
学習目標	神様の評価値との 誤差最小化	棋譜の手との 一致率最大化	報酬最大化
棋譜の要否	不要	（大量に）必要	不要
対戦の要否	不要	不要	必要 （自己対戦で可）
学習局面	任意 （ランダムに生成可）	棋譜中の局面	対戦中の局面
詰み・必至の概念	直接的に理解	間接的に理解？	間接的に理解？

「神様の評価関数」とは？

- 勝敗を完璧に予想できる評価関数のことです
- 任意の局面pに対して、勝ち=1, 負け=0というような確定的な値を返します

$$GOD(p) = \begin{cases} 1 & (win) \\ 0 & (lose) \end{cases}$$

神様の評価値をどうやって得るか？

- 仮にミニマックス探索で勝敗を読み切ることができれば、神様の評価値が得られる
- すなわち、局面pにおける神様の評価値は、局面pをルート局面として、深さを無限大でミニマックス探索を行った結果と同一になる

$$GOD(p) = search(p, depth = \infty)$$

Bishop Learningの定式化(1)

- 局面集合P={p_0, p_1, …, p_n}が与えられたとき、評価関数f(p, v)の評価値と、神様の評価関数GOD(p)の評価値との二乗誤差の総和L(P)を最小化するような特徴ベクトルvを求める

$$L(P,v) = \sum_i^n \left(f(p_i,v) - GOD(p_i) \right)^2$$

$$= \sum_i^n \left(f(p_i,v) - search(p_i, depth = \infty) \right)^2$$

Bishop Leaningの定式化(2)

- とはいえ、深さを∞にして探索することは実際上不可能なので、一定の深さDでの探索結果をもって、神様の評価関数を近似します

$$GOD(p) \approx search(p_i, depth = D, v)$$

$$L'(P,v) = \sum_{i}^{n} \left(f(p_i,v) - search(p_i, depth = D, v) \right)^2$$

- このL’(P,v)を最小化する特徴ベクトルvを求めることが、Bishop Learningの目標です

Bishop Learningの性質

- 勝ち負けを読み切れない場合、教師値（探索の結果得られる評価値）は、特徴ベクトルvに依存することになります

$$GOD(p) \approx GOD'(p,v) = search(p_i, depth = D, v)$$

- そのため、
  - 勝敗を読み切れる局面は、「教師あり」学習
  - 勝敗が読み切れない局面は、「教師なし」学習

  という性質をもち、
- 全体としては「半教師付き」学習になります

今回のBishop Learningの実装(1)

- 評価関数f(p, v)の形は？

$$f(p,v) = sigmoid(a \times g(p,v))$$

  - a : スケール・ファクター
  - g(p,v): 線形和の評価関数
  - sigmoid(x) = 1 / (1 + exp(-x))
  - スケール・ファクターとシグモイド関数を用いて、線形和の評価関数を勝率予想関数に変換
  - スケール・ファクターを導入することで、駒割の総和を一定にしても勝率への変換が正しくできるようになるため、拘束条件が課せるようになり、学習が安定化。

今回のBishop Learningの実装(2)

- 局面集合Pはどうやって用意する？
  - 乱数を使い、ランダムに局面を生成します
  - 棋譜は使っていませんが、使うことも可能だと思います
- 目的関数の最適化手法は？
  - 勾配法による最適化を使います
  - 特徴ベクトルの更新式：保木氏の手法と同じです

$$v_i^{(n+1)} = v_i^{(n)} + h\,\mathrm{sgn}\left[\frac{\partial L'(P,v)}{\partial v_i}\right]$$

Bishop Leaningの流れ

ランダムに局面集合Pを作成

探索により神様の評価値の近似値を得る

近似値を用いて目的関数の勾配を計算し、
特徴ベクトルを更新する

Bishop Learningの擬似コード

```
while (収束するまで) {
    Position samples[NUM_SAMPLES]; float weight[NUM_FEATURES];
    generateSamples(samples); // 局面集合Pを作成
    for (i = 0; i < NUM_SAMPLES; i++) {
        teacher = search(samples[i], weight); // 探索で教師値を得る
        score = evaluate(samples[i], weight); // 評価関数による評価
        for (j = 0; j < NUM_FEATURES; j++)
            gradient[j] += teacher - score; // 勾配の更新
    }
    for (i = 0; i < NUM_FEATURES; i++)
        weight[i] += h * sgn(gradient[i]); // 特徴ベクトルの更新
}
```

学習結果の例(1): 駒割

	歩	銀	金	角	飛	と	全	馬	龍
初期値	500	500	500	500	500	500	500	500	500
学習結果	117	386	401	472	523	506	420	771	904

- 知識ゼロからそれらしい結果が得られた！
  - 初期値がゼロになっていないのは、駒割の総和を拘束条件とする実装になっているためです
  - なお、頻繁に学習と実験を繰り返しているため、大会で用いる値とは異なる可能性があります

学習結果の例(2): KKP

▲5五玉、△1一玉に対する▲飛の価値

201	130	89	−27	△玉
73	105	44	56	−206
−47	−33	25	128	−39
−12	30	17	125	14
▲玉	−93	−17	77	−76

大駒は離して打て？

▲5五玉、△1一玉に対する▲角の価値

−39	−64	−22	11	△玉
−32	−55	105	−7	35
−9	141	30	9	−32
115	33	6	−29	−200
▲玉	55	46	−61	−33

角が向かい合うのを
嫌っている？

学習結果の例(3): KKP

▲5五玉、△1一玉に対する▲金の価値

−82	−146	71	−57	△玉
−7	−41	142	8	−62
−114	−75	15	62	51
24	142	−142	−68	−36
▲玉	150	12	−100	−59

金は攻守に活躍？

▲5五玉、△1一玉に対する▲銀の価値

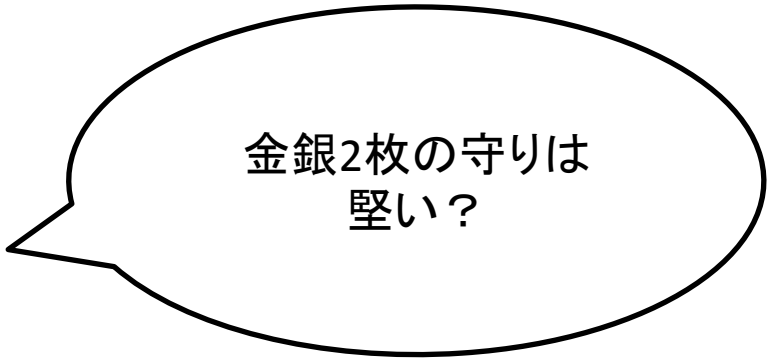
−54	−14	95	8	△玉
0	−31	11	111	49
−16	−39	40	135	86
5	−52	−21	−25	2
▲玉	3	10	−37	−90

銀は攻める駒？

学習結果の例(4): KPP

▲5五玉、▲4五金に対する▲銀の価値

−110	−28	−73	−136	28
−125	11	−91	−26	29
−94	110	−122	81	−102
295	263	371	58	−77
▲玉	▲金	39	−36	−23



さらなる発展の可能性

- 教師値は神様の評価関数の近似になっていればどんなものでもよいので、詰みや必至に限らず、様々なデータを与えることで、さらに性能を向上できる可能性があります
- 例えば…
  - 序盤であれば、定跡データベースに基づく勝率
  - 中盤であれば、モンテカルロ・シミュレーションによる平均勝率

既存の学習手法との比較（再掲）

	Bishop Learning	Bonanza Method	TD(λ)
カテゴリ	半教師付き学習 （semi-supervised）	教師あり学習 (supervised)	教師なし学習 (unsupervised)
学習目標	神様の評価値との 誤差最小化	棋譜の手との 一致率最大化	報酬最大化
棋譜の要否	不要	（大量に）必要	不要
対戦の要否	不要	不要	必要 （自己対戦で可）
学習局面	任意 （ランダムに生成可）	棋譜中の局面	対戦中の局面
詰み・必至の概念	直接的に理解	間接的に理解？	間接的に理解？
発展可能性	棋譜DBや、モンテカルロ・シミュレーションとの併用	？？	？？

謝辞

- 最後までご覧いただきありがとうございます！
- 当日もよろしくお願いします！