Opponent Modelling in Poker

Ayush Jain

Souraj Mishra

Mentor: Prof. Amitabha Mukharjee

April 15, 2013

Abstract

Poker is an interesting and a challenging problem and it is ideal for testing automated reasoning under uncertainty. The game of Poker contains both element of chance and imperfect information. Significant progress have been made regarding this but it is still a challenge to create a world class computer player to challenge best human players.

In this challenge modelling opponent strategy has always been important. Bluffing and Sandbagging are among the other important challenges. Modelling of opponent strategy can't be done deterministically and we get best results by using probabilistic framework.

Motivation

The game of poker has a rich history and provides an interesting challenge for AI researchers. It is a game of imperfect knowledge, where multiple competing agents must deal with risk management, agent modelling, unreliable information and deception, much like decision-making applications in the real world. In most games like chess our strategy doesn't depend on other player and we play our best response assuming other player to be rational but in Poker method to exploit weak opponent depends on type of mistakes each opponent makes.

Related Work

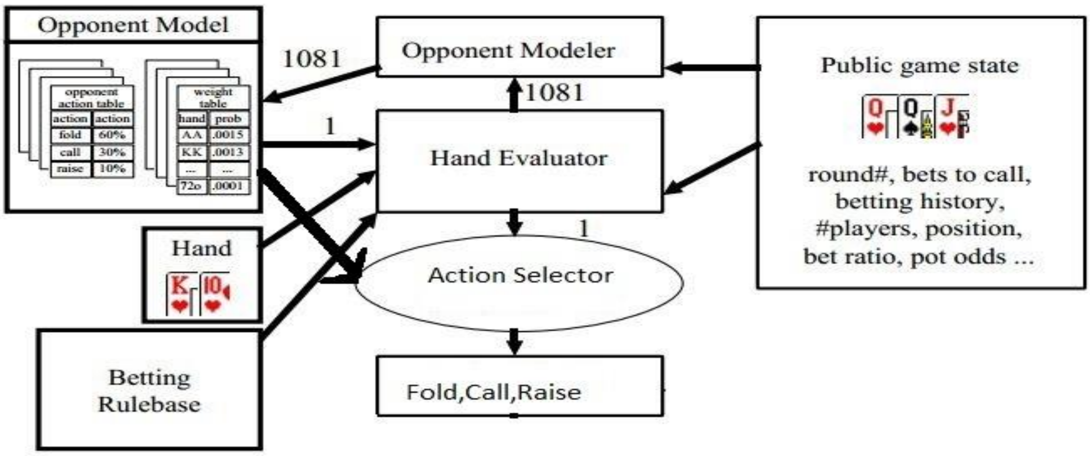
Work has also been done on 5 card game other than Texas Hold'em which is not that much complicated as it. Initial work on poker by Nicolas was not about achieving high success rates in Game but about modelling human cognitive Process. Reinforcement learning, Neural networks and other approaches also been used for opponent modelling. We will use probabilistic framework base learning of opponent strategy in this work.

Approach

We will divide the whole model of our poker playing programme into four subparts.

- Pre-Flop Evaluation
- Hand Strength And Hand Potential
- Betting Strategy
- Opponent Modelling

In next sections we will describe above parts in details.



Basic Model of Poker Playing Program

Pre-flop Evaluation

In the game of poker initially two cards are dealt to each player and no private cards are dealt. So there are 1326 {52 choose 2} possible pairs a player may hold but all of them are not distinct in terms of potential until public cards are dealt and can be reduced to 169 {(13choose2)*2+13} distinct hand types.

We use the table of these 169 hands where for each hand income rate (approximated income rate)   for different no. of opponents(2 to 9) is available which can be calculated using Gibbs sampling and available on net.

Winning percentage table for sample hands is in Fig 1.

Opponents:	1	2	3	4	5	6	7	8	9
AA	85.3	73.4	63.9	55.9	49.2	43.6	38.8	34.7	31.1
KK	82.4	68.9	58.2	49.8	43.0	37.5	32.9	29.2	26.1
QQ	79.9	64.9	53.5	44.7	37.9	32.5	28.3	24.9	22.2
Aks	67.0	50.7	41.4	35.4	31.1	27.7	25.0	22.7	20.7
AQs	66.1	49.4	39.9	33.7	29.4	26.0	23.3	21.1	19.3
JJ	77.5	61.2	49.2	40.3	33.6	28.5	24.6	21.6	19.3
KQs	63.4	47.1	38.2	32.5	28.3	25.1	22.5	20.4	18.6
AJs	65.4	48.2	38.5	32.2	27.8	24.5	22.0	19.9	18.1
KJs	62.6	45.9	36.8	31.1	26.9	23.8	21.3	19.3	17.6
ATs	64.7	47.1	37.2	31.0	26.7	23.5	21.0	18.9	17.3
AKo	65.4	48.2	38.6	32.4	27.9	24.4	21.6	19.2	17.2
TT	75.1	57.7	45.2	36.4	30.0	25.3	21.8	19.2	17.2
QJs	60.3	44.1	35.6	30.1	26.1	23.0	20.7	18.7	17.1
KTs	61.9	44.9	35.7	29.9	25.8	22.8	20.4	18.5	16.9
QTs	59.5	43.1	34.6	29.1	25.2	22.3	19.9	18.1	16.6
JTs	57.5	41.9	33.8	28.5	24.7	21.9	19.7	17.9	16.5
99	72.1	53.5	41.1	32.6	26.6	22.4	19.4	17.2	15.6
AQo	64.5	46.8	36.9	30.4	25.9	22.5	19.7	17.5	15.5
A9s	63.0	44.8	34.6	28.4	24.2	21.1	18.8	16.9	15.4
KQo	61.4	44.4	35.2	29.3	25.1	21.8	19.1	16.9	15.1

Fig 1 Winning percentage for various hands

Hand Strength and Hand Potential

After the first round of betting 3 public cards (Flop cards) are revelled.

First we will explain the naïve approach to play the poker. We calculate the probability of holding best hand given the three flop cards assuming that opponent is holding one of the {47 choose 2} different pairs with equal probability. We termed this probability as “Hand Strength”.

We can't play solely on the basis of Hand Strength as there are two more public cards being opened. Our card may be ahead initially but it may change after remaining two more cards is revelled.

So we will define two more terms as follows:

Hand Potential

Positive Potential (Ppot) - probability of improving when we are behind

Negative Potential (Npot) -probability of falling behind when we were ahead

We will calculate Effective Hand Strength (EHS) as follows

$$EHS=HS(1-Npot)+(1-HS)\ Ppot$$

This EHS is basis of for our betting strategy.

5 Cards	7 Cards			
	Ahead	Tied	Behind	Sum
Ahead	449005	3211	169504	621720
Tied	0	8370	540	8910
Behind	91981	1036	346543	439560
Sum	540986	12617	516587	1070190 = 1081x990

Ad,Qc /3h, 4c , Jh

Calculation for above sample game: HS=621720+(8910*.5)/1070190=0.59
Ppot= 91981/(439560+8910*0.5)=0.20
Npot= (169504+540*.5)/(621720+8910*.5)=0.27
EHS= 0.5127

Betting Strategy

In a betting round suppose **p** is the size of pot (money already in pot) and **b** be the size of bet we need to put to stay in game. Then pot odds is defined as **b/b+p.**

Define **d** as : d=EHS -(b/(b+p))

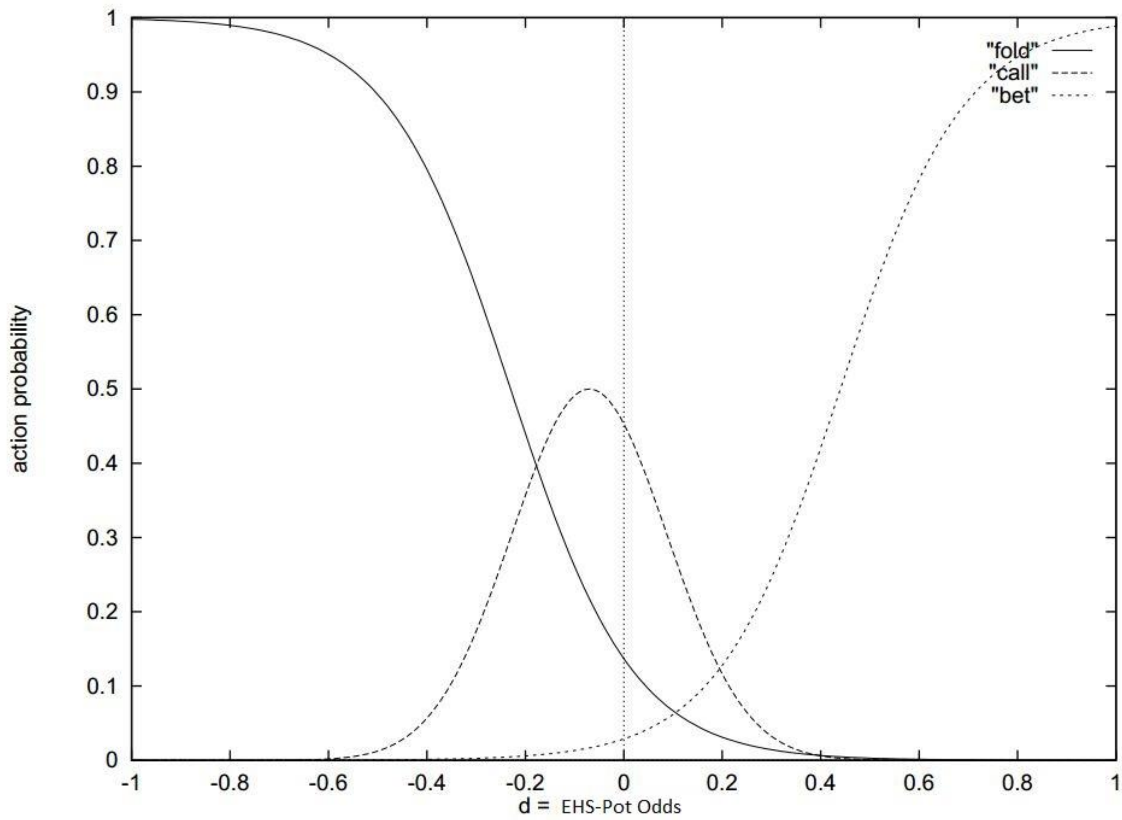
In a naïve approach we could have sat deterministic boundary on d for Raise, Fold and call. But that strategy can be exploited by opponent and we can't do anything in case of Bluffing and Sandbagging. So we will rather use probabilistic method for batting mentioned below:

Bet Prob=1/(1+exp(-a(d-f1)))

Fold prob=1/(1+exp(a(d+f2))

Call prob=exp(-20(d+fc)^2)

Values of various constant can be varied according to stereotype of opponent determined by history of the hands for the opponent.



Betting Curves

Opponent Modelling

In most of the games like chess opponent modelling is not that important as we play our best response irrespective of opponent's strategy but it is not the case with poker where outcome of game heavily depends on the way opponent plays. So Opponent Modelling has always been important and a challenging problem.

We will divide this challenge of opponent modelling in three subparts.

Weighing the Enumeration: In the naïve approach discussed so far we have assumed opponent is holding each possible pair with equal probability but it is not the case as if he has cards with week potential then he would have folded the cards in earlier round. To account this it is proposed to used the weight which represents the probability that if opponent got that card he have not folded yet given actions taken by both players and cards opened till now and past history. Weights are recomputed based on opponent actions. Each time player makes an action weights are modified.

Computing Initial Weight: Initial weights are based on the history of game and opponent's actions. We calculate median and variance of income rate of folding, raising and calling hand. And then use linear interplotetion around median with median having weight equal to 0.5.

Reweighing: We calculate the median hand required for an action. And do linear interplotation around that to calculate reweighing factor. For reweighing we multiply these factors with the weights to get new weights and do hand calculations based on these.

Hand	Weight	HR	HS_1	$\sim PP_2$	EHS	Rwt	Nwt	Comment
Jc 4h	0.01	0.993	0.99	0.04	0.99	1	0.01	very strong, but unlikely
Ac Jc	1	0.956	0.931	0.09	0.94	1	1	strong, very likely
5h 2h	0.2	0.004	0.001	0.35	0.91	1	0.2	weak, but very high potential
6s 5s	0.6	0.026	0.006	0.21	0.76	0.9	0.54	weak, good potential
5s 5h	0.7	0.816	0.736	0.04	0.74	0.85	0.6	moderate, low potential
5s 3s	0.4	0.648	0.671	0.1	0.7	0.75	0.3	mediocre, moderate potential
Ac Qd	1	0.585	0.584	0.11	0.64	0.6	0.6	mediocre, moderate potential
7s 5s	0.6	0.052	0.012	0.12	0.48	0.2	0.12	weak, moderate potential
Qd Td	0.9	0.359	0.189	0.07	0.22	0.01	0.01	weak, little potential

Ad,Qc /3h, 4c , Jh

Example of reweighing calculations based

on opponent model (from[1])

Implementation

We have implemented the game of Poker where user can play against the Poker playing Programme which follows the described strategy

Future work

Successful opponent modelling against Human is still a challenge. Above works when opponent strategy doesn't vary much with time. No. of games required for above specified strategy to learn modelling is not of practical use as we don't get same opponent to play that much no. of games. Bayesian network can be used to model uncertainty in game as well as opponent.

References

[1] D. Billings, D. Papp, J. Schaeffer, D. Szafron ,Opponent modeling in poker

Proc. AAAI-98, Madison, WI (1998), pp. 493–499

[2] D. Billings, A. Davidson, J. Schaeffer, D. Szafron ,The challenge of poker

Artificial Intelligence, 134(1–2):201–240, 2002.

[3] F. Southey, M. Bowling, B. Larson, C. Piccione, N. Burch, D. Billings, and C. Rayner.

Bayes' bluff: Opponent modelling in poker. In 21st Conference on Uncertainty in

Artificial Intelligence, UAI'05)

[4] D. Sklansky, M. Malmuth

Hold'em Poker for Advanced Players (2nd Edition)Two Plus Two Publishing (1994)

[5] Kevin B. Korb, Ann E.Nicholson and Nathalie Jitnah, Baysian Poker

[6] Cactus Kev's Hand evaluator implementation in C++

[7] Daniel Scocco , Poker framework Source Code