Most Important Fundamental Rule of Poker Strategy

Sam Ganzfried, Max Chiswick
sam@ganzfriedresearch.com, max.chiswick@gmail.com

Abstract

Poker is a large complex game of imperfect information, which has been singled out as a major AI challenge problem. Recently there has been a series of breakthroughs culminating in agents that have successfully defeated the strongest human players in two-player no-limit Texas hold 'em. The strongest agents are based on algorithms for approximating Nash equilibrium strategies, which are stored in massive binary files and unintelligible to humans. A recent line of research has explored approaches for extrapolating knowledge from strong game-theoretic strategies that can be understood by humans. This would be useful when humans are the ultimate decision maker and allow humans to make better decisions from massive algorithmically-generated strategies. Using techniques from machine learning we have uncovered a new simple, fundamental rule of poker strategy that leads to a significant improvement in performance over the best prior rule and can also easily be applied by human players.

Introduction

Poker is an extremely challenging game, both for computer AI programs and for humans, due to the large state spaces and imperfect information. Poker has been very popular with humans for centuries, with the most common variant in recent years being no-limit Texas hold 'em. There are two primary reasons for the popularity of poker in comparison to other games. First is that, while being predominantly a game of skill (which allows professional players to achieve long-term success), there is still a significant degree of luck and variance. In a game like chess, a grandmaster player will beat a recreational club-level player 100% of the time; however in poker, even an amateur or novice player can still win a single hand or session even against a strong professional player (in fact the Main Event at the World Series of Poker in Las Vegas has been won several times by amateur players). The second reason is that the rules are very simple (allowing easy accessibility to novice players), yet perfect mastery is extremely complex. As a famous professional player Mike Sexton stated, "The name of the game is No Limit Texas hold 'em, the game that takes a minute to learn but a lifetime to master." While anyone can understand the rules and start playing very quickly, even the best experts spend many years (for some an entire lifetime) learning and improving.

Humans learn and improve in a variety of ways. The most obvious are reading books, hiring coaches, poker forums and discussions, and simply playing a lot to practice. Recently several popular software tools have been developed, most notably PioSolver,[1] where players solve for certain situations, given assumptions for the hands the player and opponent can have. This is based on the new concept of *endgame solving* [Ganzfried and Sandholm, 2015], where strategies are computed for the latter portion of the game given fixed strategies for the *trunk* (which are input by the user). While it has been pointed out that theoretically this approach is flawed in the worst case in imperfect-information games (such as poker) [Ganzfried and Sandholm, 2015], nonetheless it has played a pivotal role in producing superhuman agents for two-player no-limit Texas hold 'em [Brown and Sandholm, 2017a; 2017b; 2017c; Moravčík *et al.*, 2017]. Human players can improve from training with such software tools by studying solutions to many different scenarios, but this requires a lot of time and effort, ability to correctly gauge the trunk strategies, and a significant amount of technical savvy.

Learning to improve as a human is complicated further as there are many different variants of poker commonly played, and concepts learned from one may not transfer over to the other. However, certain theoretical concepts are fundamental to poker and can extend throughout different variants, independent of the specific situation (and in some cases even the opponent). For example, the concept of *pot odds* dictates how good one's *equity* needs to be to make a call worthwhile. For example, suppose the opponent bets $10 into a pot that initially has $30. If we call and win then we will profit $10 + $30 = $40, and if we call and lose then we profit -$10. By contrast, note that if we fold to the bet then we will profit $0. Suppose we think that we have the best hand with probability p: then we are indifferent between calling and folding if $40p - 10(1 - p) = 0 \leftrightarrow p = \frac{1}{5}$. So calling is more profitable than folding if we think that we win over 20% of the time, and folding is more profitable if we think that we win less than 20% of the time. This is a clear simple principle that is

---

applicable to all variants of poker and can be easily applied by human players. (Note however that this concept is not a *rule* that specifies what action the player should take on an absolute level; a *range* of hands must be specified for the opponent in order to apply the concept.) While mastering poker is clearly much more challenging than learning a few general concepts such as pot odds, nonetheless certain fundamental rules grounded in math and game theory can be very important, particularly at earlier phases of study.

Pot odds does not dictate a full rule for how one should play against opponents with potentially unknown strategy. The most widespread rule for this is based on the concept of the *Minimum Defense Frequency* (MDF). This concept, which is often denoted by α [Ankenman and Chen, 2006], specifies the probability needed to call when facing a bet of a certain size in order to ensure that the opponent cannot profitably bluff us with his weakest hands. (It also simultaneously specifies the optimal ratio that a player should use of *value bets* (bets with a strong hand) to *bluffs* (bets with a weak hand) that a player should use.) The minimum defense frequency is derived from the Nash equilibrium of a simplified game, and has been singled out as one of the most important concepts of poker. When facing a bet by a player with an unknown strategy, many strong players will base their strategy off of MDF. This is a simple effective rule that is easily understood by humans and applies to many different variations of poker. In this paper we will develop a new rule that is also simple and can easily be understood by humans that improves performance significantly beyond MDF. This rule integrates MDF with another popular and easily understandable poker concept called *Range Advantage*, and is derived by applying machine learning algorithms to a database of Nash equilibrium solutions to simplified games. We believe that this is the most important single rule for strong poker play.

Poker is large game of imperfect-information and has been singled out as being a major AI challenge problem. Recently programs for two-player no-limit Texas hold 'em have competed against the strongest human players in a sequence of high-profile "Brains vs. Artificial Intelligence" competitions [Brown and Sandholm, 2017c] culminating in the agent Libratus that won decisively in 2017. Independently a second agent DeepStack was also created that defeated human players in 2017, though the humans were not experts in the specific game format [Moravčík *et al.*, 2017]. Despite these successes, there have been relatively few takeaways from the research that aspiring human players can readily apply to improve their poker game; while several of the hands from these competitions are publicly available, it is difficult to construe strategic insights that will generalize from just observing several isolated situations. In a separate new line of research, a new approach has been developed using decision trees to compute strategic rules that are easily understood by humans [Ganzfried and Yusuf, 2017]. This has led to deduction of new rules concerning when a player should make an extremely small bet and an extremely large bet. We continue this avenue of research by exploring for the first time the connection between two key poker strategy concepts, producing a single understandable rule that applies to a variety of bet sizes and different game variations and significantly improves performance over the popular MDF concept. Large-scale computation of strong game-theoretic strategies is also important in many domains other than poker; for example, it has been applied recently to randomized security check system for airports [Paruchuri *et al.*, 2008]. Approaches for computing human-understandable rules are critical for situations such as national security where humans will be making important real-time decisions.

No-limit poker

We will be studying the following simplified poker game. First player 1 is dealt a card from a n-card deck (cards are the numbers 1–n), according to a probability distribution p (e.g., p_2 is the probability that player 1 is dealt a 2). We assume similarly that player 2 is dealt a card from 1–n according to q. We assume the cards are independent, and that it is possible for both players to be dealt the same card (so that our experiments will not be confounded by the additional issue of *card removal*). Players initially start with a stack S and there is a pot of size P, with a deck size equal to $n = 10$. Then player 1 can either *check* or *bet* a size in $\{b_{1i}\}$. Facing a bet from player 1, player 2 can then call or *fold* (forfeit the pot). Facing a check from player 1, player 2 can check or bet a size in $\{b_{2i}\}$. Then player 1 can choose to call or fold facing a bet. If neither player folds, the player with higher card wins the pot.

For our initial experiments we will assume the pot and stack sizes P and S both equal 1 (if they differed we could divide by the size of the pot to normalize). For both players we only allow a bet size equal to pot (1) for b_{1i}, b_{2i}. We will also consider settings where the bet size is equal to 0.5 times the pot and 0.75 times the pot (which are popular human betting sizes). Of course this setting can allow arbitrary values for these parameters and multiple bet sizes, but for concreteness we decided to start with these values.

It is clear that the optimal strategies (according to Nash equilibrium) for the two players will vary significantly depending on the distributions p and q. For example, if both players are dealt cards uniformly (Figure 1), then a Nash equilibrium strategy for player 1's first action is:

- Card 1: Check pr. 1
- Card 2: Check pr. 0.5, bet 1 pr. 0.5
- Card 3-8: Check pr. 1
- Card 9: Bet 1 pr. 1
- Card 10: Check pr. 1

This can be computed quickly using, e.g., a linear programming formulation [Koller and Megiddo, 1992]. Note that the equilibrium strategy includes *bluffing* (i.e., betting with weak hands such as 2), as well as *slowplaying*, aka *trapping* (checking with a strong hand such as 10).

However, suppose the cards are dealt according to a different distribution: player 1 is either dealt a very strong hand (10) or a very weak hand (1) with probability 0.5 while player 2 is always dealt a medium-strength hand (Figure 2). Then the equilibrium strategy for player 1 is:

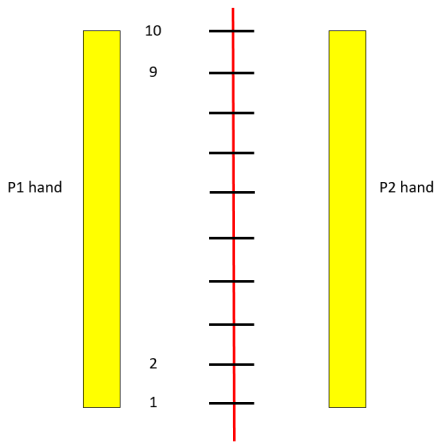- Card 1: Bet 0 pr. 0.5, 1 pr. 0.5
- Card 10: Bet 1 pr. 1

Figure 1: Example game where both players are dealt private information uniformly at random over all hands.
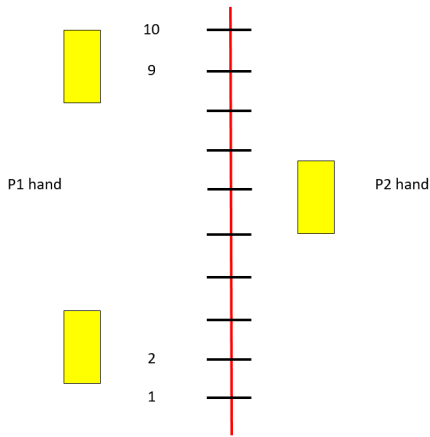

Figure 2: Player 1 is dealt either a very strong or very weak hand and player 2 is always dealt a mediocre hand.
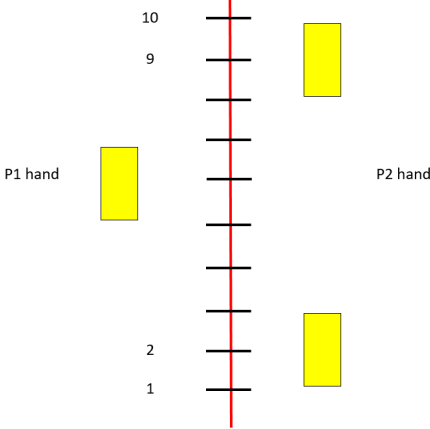

Figure 3: Player 1 is always dealt a mediocre hand and player 2 is either dealt a very strong or very weak hand.

If player 1 is always dealt a medium-strength hand (5) while player 2 is dealt a very strong or very weak hand with probability 0.5 (Figure 3), then the equilibrium strategy is:

- Card 5: Bet 0 pr. 1

Qualitative models and endgame solving

There has been some prior study of human understandable strategies in imperfect-information games, and in poker specifically. Ankenman and Chen compute analytical solutions of several simplified poker variants by first assuming a given human-understandable qualitative structure on the equilibrium strategies, and then computing equilibrium strategies given this presumed structure, typically by solving a series of indifference equations [Ankenman and Chen, 2006]. While the computed strategies are generally interpretable by humans, the models were typically constructed from a combination of trial and error and expert intuition, and not constructed algorithmically. More recent work has shown that leveraging such qualitative models can lead to new equilibrium-finding algorithms that outperform existing approaches [Ganzfried and Sandholm, 2010]. That work proposed three different qualitative models for the final round of two-player limit Texas hold 'em (Figure 4), and showed empirically that equilibrium strategies conformed to one of the models for all input information distributions (and that all three were needed). Again here the models were constructed by manual trial and error, not algorithmically.

We note that while the problem we are considering in this paper is a "toy game," it captures important aspects of real poker games and we expect our approaches to have application to larger more realistic variants (our game generalizes many common testbed variants; if the deck had only 3 cards and only one bet size is allowed, this would be the commonly-studied variant "Kuhn poker" [Kuhn, 1950]). In the recent Brains vs. Artificial Intelligence two-player no-limit Texas hold 'em competition, the agent Claudico computed the strategy for the final betting round in real time, and a top human player has commented that the "endgame solver" was the strongest component of the agent [Ganzfried, 2015], and endgame solving was also a crucial component of subsequent success of the improved agent Libratus [Brown and Sandholm, 2017b]. Another recent superhuman agent DeepStack can also be viewed as applying endgame solving, as it works by viewing different rounds of the game as separate endgames which are solved independently, using deep learning to estimate the values of the endgames terminal states [Moravčík et al., 2017]. Endgame solving assumes that both agents had private information distributions induced by the strategies for the prior rounds using Bayes' rule, assuming they had been following the agent's strategy for the prior rounds [Ganzfried and Sandholm, 2015]. The game we study here is very similar to no-limit Texas hold 'em endgames, except that we are assuming a ten-card deck, specific stack and betting sizes, and that raises are not allowed. We expect our analysis to extend in all of these dimensions and that our approaches will have implications for no-limit Texas hold 'em and many other popular variants that may be more complex.
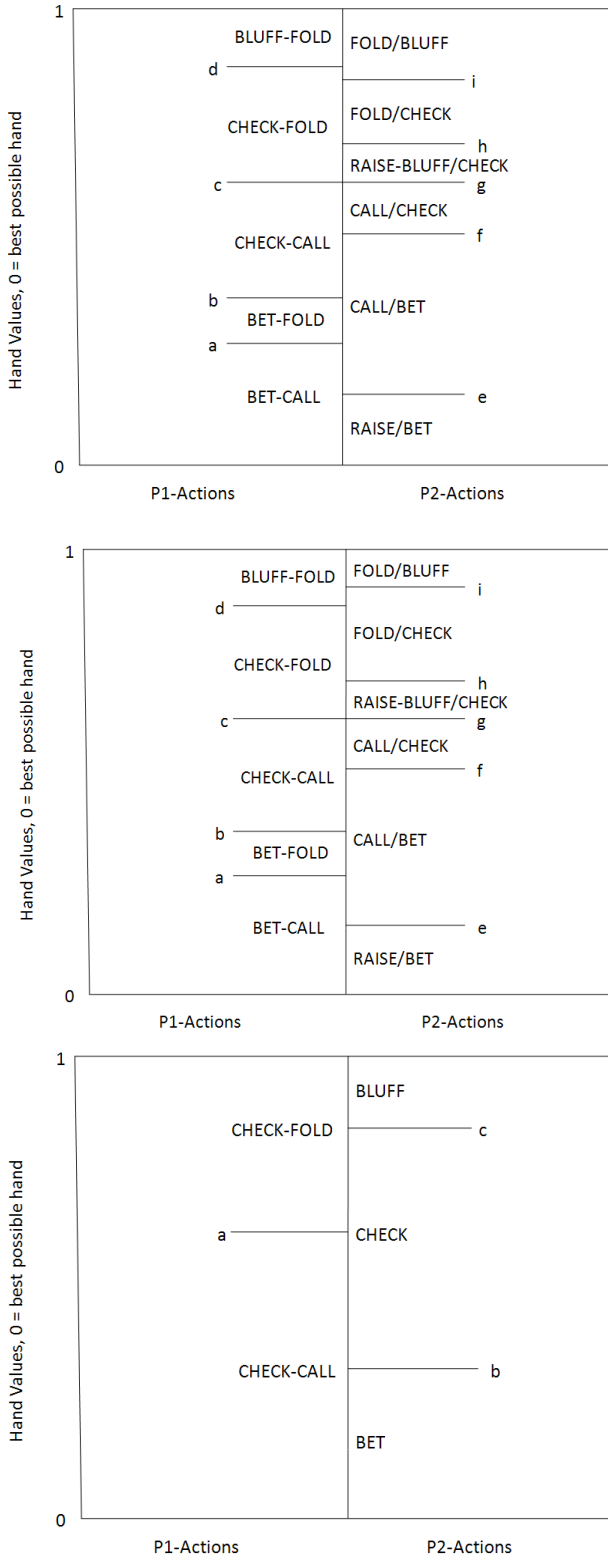
Figure 4: Three qualitative models for two-player limit Texas hold 'em river endgame play.

Minimum defense frequency and range advantage

For a bet of size b and a pot of size P, the *minimum defense frequency* (MDF) is defined to be the quantity $\alpha = \frac{P}{b+P}$. Suppose player 2 calls the bet with probability α, and suppose that player 1 has an extremely weak hand that he is contemplating betting as a bluff. With probability α the bluff would be called and player 1 would lose b, while with probability $1 - \alpha$ player 2 would fold and player 1 would win P. Thus the expected profit for player 1 from betting is:

$$\alpha\cdot(-b)+(1-\alpha)P = P-\alpha(b+P) = P-\frac{P}{b+P}\cdot(P+b) = 0.$$

Thus player 1 is precisely indifferent between betting as a bluff and not betting with his weak hand. If player 2 were to call with any probability below α, then player 1 would be able to have a guaranteed profit by bluffing with all his weak hands; therefore, the value α is the minimum amount of the time that player 2 must call (or raise if it is permitted) against a bet in order to prevent player 1 from profitably betting with all his weak hands. For our parameters of $b = 1, P = 1$ the MDF is $\alpha = \frac{1}{2}$. Note also that if player 2 calls with probability exceeding α, then player 1 would never want to bluff with his weak hands (in which case player 2 would never want to call with his medium-strength hands, etc.). So in general we would expect player 2 to call with frequency precisely equal to α; however this is not always the case. Consider an extreme example when player 1 is always dealt a 10 and player 2 is always dealt a 1. Then clearly player 2 would always prefer to fold to a bet because he will never have the best hand. Note that in this example player 1 has a *range advantage* equal to 1. In general, if player 1 has a significant range advantage, optimal play (i.e., according to Nash equilibrium strategies) may suggest that player 2 call with probability below α.

The *range advantage* for player 1 under the distributions p, q is player 1's *equity* in the hand given the distributions. This is equal to the probability that player 1 will have a better hand under p plus one-half times the probability that they have the same hand. So we define the Range Advantage as:

$$\text{Range Advantage}(p, q) = \left(\sum_{j=1}^{n}\sum_{i=j+1}^{n} p_iq_j\right) + \left(\frac{1}{2}\sum_{i=1}^{n} p_iq_i\right) \tag{1}$$

We generated 100,000 games with uniform random distributions for p and q (recall that p is player 1's distribution and q is player 2's). We then solve these games for a Nash equilibrium. Let $c(i)$ be the optimal frequency for player 2 to call a bet from player 1 with card i according to the equilibrium strategy. Then the *optimal defense frequency* (ODF) c^* is the weighted sum of the call probabilities over all hands, i.e., $c^* = \sum_i \left(q_i c(i)\right)$. In our experiments we seek to understand the relationship between the range advantage and the optimal defense frequency which, as described above, may differ considerably from the minimum defense frequency value.

For each hand in the sample, we can add a data point where the x coordinate is the range advantage of the distributions p, q, and the y coordinate is the optimal defense frequency c^*. A scatterplot of points selected according to this process is shown in Figure 5. The graph depicts a clear negative trend

between the range advantage of player 1 and the optimal defense frequency for player 2. We also observe a mass of hands with optimal frequency exceeding the MDF value of 0.5. We note that each game may contain several solutions, and it is possible that there also exist other equilibria with different ODF values. Thus, for the games with ODF exceeding 0.5, it is possible that the game also contains an equilibrium with ODF equal to 0.5, and that our algorithm happened to select this one. This situation can arise for games where player 1's equilibrium strategy has him always check with all of his hands and never bet; in this case there are many possible equilibrium strategies for player 2 to play in response to a hypothetical "off-path" bet action by player 1, and player 2 need only play a strategy that does not give player 1 incentive to deviate and start betting some hands instead of checking.
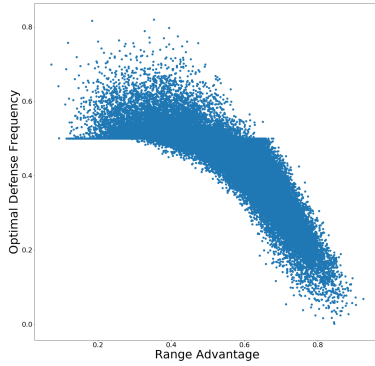

Figure 5: Range advantage vs. optimal defense frequency over 100,000 random games with bet size equal to pot.

Experiments

In the first set of experiments, we compare several approaches for the variant where the players are restricted to just the pot-sized bet: "previous expert knowledge [has] dictated that if only a single bet size [in addition to all-in] is used everywhere, it should be pot sized" [Hawkin *et al.*, 2012].

For each approach, we compute the mean-squared error (MSE) from the optimal defense frequency (ODF) over the 100,000 game sample (recall that MSE is the average of the square of the difference between the predicted and actual value). First we consider just predicting the MDF, which is 0.5 for this case of a pot-sized bet, which produces MSE of 0.0067. We next considered "Linear MDF," which in this case would select the optimal constant predictor, which selected 0.466 as opposed to 0.5, and produced MSE of 0.0055. We next integrated range advantage as a feature to produce the optimal regressor prediction of -.446*RA + 0.688, with MSE 0.0024, and finally quadratic regression which produced MSE of 0.0012. These results (Table 1) show that adding in range advantage as a feature leads to a significant reduction in MSE over just using MDF (around a 56% reduction in MSE even accounting for optimized normalization of MDF), while also unsurprisingly quadratic leads to a further reduction. Though there is a risk of overfitting, the plot in Figure 5 does appear to fit closer with a quadratic than a linear curve, so the quadratic

predictor may be more appropriate. For all approaches we compute average MSE over 10-fold cross validation.

Approach	Formula	MSE
Fixed MDF	0.5	0.006686
Linear MDF	0.466	0.005512
Linear RA	-.446*RA + 0.688	0.002406
Quadratic RA	-1.760*RA^2 + 1.316*RA + 0.275	0.001161

Table 1: Results for simplified game with pot bet size.

We next ran experiments over a larger dataset consisting of 100,000 games for each of the three bet sizes of 0.5 pot, 0.75 pot, and pot (so the dataset consisted of solutions of 300,000 total games). These experiments allowed us to uncover a rule that would extend beyond different bet sizes, as the different bet sizes correspond to different values of the MDF α. For these experiments the MDF now becomes a feature (while it was just the constant 0.5 for the prior experiments). The MSE of several approaches are given in Table 2. The MSE of fixed MDF is 0.0086, which is reduced to 0.0068 for linear MDF (note that, in contrast to the pot-sized experiments where MDF was a fixed constant, in this setting where MDF is a feature there is now both a coefficient for MDF and a constant parameter). Using both MDF and RA as features brought down MSE further to 0.0030 (similar to the pot-sized experiments including RA as a feature reduces MSE over linear MDF by around 56%). We also used the product of RA and MDF as a new potential feature, and including this did not lead to a significant reduction in MSE over just using RA (both when used in addition to RA and in place of it).

Note that the optimal linear predictor using RA and MDF (rounded to 3 decimal places) is 0.904*MDF - 0.495*RA + 0.261. Noting that the first coefficient is close to 1, the second is close to 0.5, and the third is close to 0.25, we also investigated the MSE of this "simplified" predictor with coefficients that can be more easily remembered and applied in real time. This produced an MSE of 0.0048, which is about halfway between the MSE of Linear MDF and Linear MDF with RA.

Another observation we made from the graph in Figure 5 is that the optimal defense frequency only rarely exceeds the MDF (and furthermore the pattern for the points where ODF exceeds MDF seems to be different from that of the points for which it doesn't). So we decided to consider an approach where we truncate the predictor at the MDF (i.e., take the minimum of the predictor value and MDF). We also considered an approach where we used a separate simple linear function for the cases where player 1 has a significant range advantage (RA > 0.75) of 1-RA for the RA values exceeding 0.75, while using the same approach as before for RA ≤ 0.75.

Using min(MDF, 0.0904*MDF - 0.495*RA + 0.261) produced MSE of 0.0026, which is only a small improvement over that of linear MDF with RA (without using the min function). However, applying the min function to the simplified linear predictor of MDF - 0.5*RA + 0.25 led to a significant reduction in MSE from 0.0048 to 0.0032 (33% reduction). Adding in the separate rule of using (1-RA) for RA > 0.75 further decreased MSE to 0.0024. Finally we note that using quadratic regression not surprisingly produced the lowest

Approach	Formula	MSE
Fixed MDF	MDF	0.008586
Linear MDF	0.904*MDF + 0.0140	0.006791
Linear MDF with RA	0.904*MDF - 0.495*RA + 0.261	0.002996
Linear MDF with RA*MDF	1.330*MDF - 0.851*[RA*MDF] + 0.0134	0.002971
Linear MDF with RA and RA*MDF	1.202*MDF - 0.150*RA - 0.595*[RA*MDF] + 0.0885	0.002966
Simplified Linear MDF with RA	MDF - 0.5*RA + 0.25	0.004795
Min Linear MDF with RA	min(MDF, 0.904*MDF - 0.495*RA + 0.261)	0.002638
Simplified Min Linear MDF with RA	**min(MDF, MDF - 0.5*RA + 0.25)**	**0.003196**
Piecewise Simp. Min Linear MDF w. RA	1 - RA, RA > 0.75; min(MDF, MDF - 0.5*RA + 0.25), RA $\leq$ 0.75	0.002446
Quadratic MDF with RA and MDF*RA	-1.945*RA^2 - 0.615*RA*MDF + 0.0111*MDF^2 + 1.806*RA + 1.197*MDF - 0.368	0.001459

Table 2: Results for several approaches over 100,000 hands for each of three games with bet sizes of 0.5 pot, 0.75 pot, and pot.

MSE out of the approaches considered with 0.0015. There are many more complex approaches we could have also experimented with (e.g., additional features, decision trees, higher degree regression, etc.), however these run a significant risk of overfitting (and also producing less-understandable rules).

New fundamental rule of poker strategy

Many of the approaches presented in Table 2 achieve a significant performance improvement over just (linear) MDF. While not the best-performing approach, we single out the rule min(MDF, MDF - 0.5*RA + 0.25) as being the best-performing single rule using simple coefficients that are easy for a human to remember and apply. This rule produces MSE of 0.0032, which is over a 50% reduction in MSE over linear MDF (and a 65% reduction in MSE over fixed MDF). This rule is a single equation, with very simple values of the coefficients (1, -0.5, and 0.25). It only performs slightly worse than the analogous approaches that use more complicated coefficient values. It is also outperformed by a piecewise approach that adds in a second case of (1-RA) for the case RA > 0.75, and uses the same predictor for RA $\leq$ 0.75. However, we consider this approach as essentially two different rules combined into one approach, not as a single rule.

Fundamental Rule of Poker Strategy 1. *Given the minimum defense frequency value MDF when facing a certain bet size, and assuming a range advantage of RA, then you should call the bet with a fraction of the hands in your range equal to min(MDF, MDF - 0.5*RA + 0.25), and fold otherwise.*

Note that for RA = 0 (player 2 always has the best hand), this will predict ODF = min(MDF, MDF + 0.25) = MDF. For RA = 1 it will predict ODF = min(MDF, MDF -0.25) = MDF - 0.25. And for RA = 0.5 (neither player has a range advantage) it will predict ODF = min(MDF, MDF) = MDF (which is expected for the case when no player has a clear advantage).

As an example to apply this rule, suppose we are in a setting where opponent bets the pot (so MDF = 0.5), and we believe that the opponent has a range advantage of 0.8. Then we have min(MDF, MDF - 0.5*RA + 0.25) = min (MDF, MDF - 0.15) = MDF - 0.15 = 0.5-0.15 = 0.35. So the rule would stipulate that we should call 35% of the time.

Recall that previously the best rule to apply here would be just to use ODF = MDF = 0.5. Our new rule produces an improvement of over 65% over the best prior approach, while being a simple rule that can be easily understood and applied

in real time. Note that had we instead defined the range advantage to range from -1 to 1 instead of 0 to 1 (so that RA of 0 meant no advantage and +1 means player 1 has full advantage), then the rule becomes min(MDF, MDF - 0.25*RA), which is even easier for a human to interpret. This is obtained by applying the transformation $RA_{\text{new}} \leftarrow 2 * RA_{\text{old}} - 1$.

Conclusion

Using techniques from machine learning, we have been able to obtain a new important rule of poker strategy that achieves significantly lower MSE than the most popular rule of basing calling decisions on MDF. We obtain the 100–50–25 MIN rule, which is a single easily-understandable equation for the predictor of ODF, which produces an improvement of over 65% of the best prior approach. Of course ever better rules could be obtained (we obtained some, for example with more complex coefficients, piecewise linear functions that branch based on RA value, and quadratic regression); however these are all more complex and would be harder for a human player to memorize and implement in real time. There can also be some concern about overfitting with more complex rules.

Potentially an even better rule could be obtained if new improved features are used (e.g., an alternative definition of Range Advantage). It would be interesting to explore implications for the recent line of work on developing approaches for computing human-understandable strategies to other domains where game-theoretic algorithms produce strategies that must be interpreted by human decision makers. For example, in national security humans often ultimately make the decisions that may be generated by algorithms [Paruchuri *et al.*, 2008].

From a poker perspective, it would be interesting to see whether we could create a portfolio of important understandable rules using machine learning, and then create a full agent for a complex poker variant such as no-limit Texas hold 'em that plays according to this portfolio. This would fall under the category of being a *rule-based system*, which generally are outperformed by more principled approaches; however, an advantage would be that the approach would require very few computational resources, in contrast to the strongest agents that require tens of millions of core hours. It would be very interesting to see how an agent based on a portfolio of simple human-understandable rules would fare against the strongest current agents that are based on running equilibrium algorithms on supercomputers to create massive strategy files.

References

[Ankenman and Chen, 2006] Jerrod Ankenman and Bill Chen. *The Mathematics of Poker*. ConJelCo LLC, Pittsburgh, PA, USA, 2006.

[Brown and Sandholm, 2017a] Noam Brown and Tuomas Sandholm. Safe and nested endgame solving in imperfect-information games. In *AAAI Workshop on Computer Poker and Imperfect Information Games*, 2017.

[Brown and Sandholm, 2017b] Noam Brown and Tuomas Sandholm. Safe and nested subgame solving for imperfect-information games. In *Proceedings of the Annual Conference on Neural Information Processing Systems (NIPS)*, 2017.

[Brown and Sandholm, 2017c] Noam Brown and Tuomas Sandholm. Superhuman AI for heads-up no-limit poker: Libratus beats top professionals. *Science*, 2017.

[Ganzfried and Sandholm, 2010] Sam Ganzfried and Tuomas Sandholm. Computing equilibria by incorporating qualitative models. In *Proceedings of the International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, 2010.

[Ganzfried and Sandholm, 2015] Sam Ganzfried and Tuomas Sandholm. Endgame solving in large imperfect-information games. In *Proceedings of the International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, 2015.

[Ganzfried and Yusuf, 2017] Sam Ganzfried and Farzana Yusuf. Computing human-understandable strategies: Deducing fundamental rules of poker strategy. *Games*, 8(4):1–13, 2017.

[Ganzfried, 2015] Sam Ganzfried. Reflections on the first man vs. machine no-limit Texas hold 'em competition. *SIGecom Exchanges*, 4.2, 2015.

[Hawkin et al., 2012] John Hawkin, Robert Holte, and Duane Szafron. Using sliding windows to generate action abstractions in extensive-form games. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2012.

[Koller and Megiddo, 1992] Daphne Koller and Nimrod Megiddo. The complexity of two-person zero-sum games in extensive form. *Games and Economic Behavior*, 4(4):528–552, Oct 1992.

[Kuhn, 1950] H. W. Kuhn. A simplified two-person poker. In H. W. Kuhn and A. W. Tucker, editors, *Contributions to the Theory of Games*, volume 1 of *Annals of Mathematics Studies, 24*, pages 97–103. Princeton University Press, Princeton, New Jersey, 1950.

[Moravčík et al., 2017] Matej Moravčík, Martin Schmid, Neil Burch, Viliam Lisý, Dustin Morrill, Nolan Bard, Trevor Davis, Kevin Waugh, Michael Johanson, and Michael Bowling. Deepstack: Expert-level artificial intelligence in heads-up no-limit poker. *Science*, 2017.

[Paruchuri et al., 2008] Praveen Paruchuri, Jonathan P. Pearce, Janusz Marecki, Milind Tambe, Fernando Ordonez, and Sarit Kraus. Playing games with security: An efficient exact algorithm for Bayesian Stackelberg games. In *Proceedings of the International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, 2008.