Game Theory

10. Poker

Bernhard Nebel and Robert Mattmüller
1 Motivation

Kuhn Poker

Real Poker: Problems and techniques

Counterfactual regret minimization
The system Libratus played a Poker tournament (heads up no-limit Texas hold ’em) from January 11 to 31, 2017 against four world-class Poker players.

- Heads up: One-on-One, i.e., a zero-sum game.
- No-limit: There is no limit in betting, only the stack the user has.
- Texas hold’em: Each player gets two private cards, then open cards are dealt: first three, then one, and finally another one.
- One combines the best 5 cards.
- Betting before the open cards are dealt and in the end: check, call, raise, or fold.

- Two teams (reversing the dealt cards).
- Libratus won the tournament with more than 1.7 Million US-$ (which neither the system nor the programming team got).
The humans behind the scene

Professional player Jason Les and Prof. Tuomas Sandholm (CMU)
2 Kuhn Poker
Kuhn Poker

- Minimal form of heads-up Poker, with only three cards: Jack, Queen, King.
- Each player is dealt one card and antes 1 chip (forced bet in the beginning).
- Player 1 can check (declines to make a bet), or bet 1 chip.
- After player 1 has checked, player 2 can check or bet. If player 2 bets, player 1 can fold or call (also betting one chip)
- After Player 1 has bet, player 2 can fold or call.
Kuhn Poker: Game tree
Kuhn has shown:

- There exist a family of Nash equilibria behavioral strategies for player 1 and one behavioral NE strategy for player 2.
- In this Nash equilibrium, the expected payoff for player 1 is $-1/18$.
- That shows the systematic disadvantage, the first player has!
3 Real Poker: Problems and techniques
Reminder: In chess, there are $10^{47}$ distinct states, in Backgammon there are $10^{20}$.

Heads-up limit Texas hold’em has $10^{17}$ distinct states and $10^{14}$ information sets.

No-limit: Depends on stack. With 20k$: $10^{161}$ information sets.
General techniques

- **Abstraction**: Action abstraction (bet size) and card abstractions (classifying similar hands into buckets) → only $10^{12}$ information sets.

- **Equilibrium computation**: Using counterfactual regret minimization as a self-play technique.

- **Sub-game solving**: In later betting rounds, one solves the game with a finer abstraction (and the information gained from the game so far).

- **Self-Improvement**: During the night, new parts of the game tree are explored, when abstraction is too coarse there.

- 25 Million core hours to compute strategies.
4 Counterfactual regret minimization
Regret matching in strategic games

Play a strategic game for a number of rounds:

- **Regret** is determined after each game round: If I had played another move, my payoff would have been *that* much higher!
- **Accumulate** all positive regrets over time.
- **Match** the probabilities of a mixed strategy with the accumulated regret.

Take the **average** over all mixed strategies.

If two players use the regret matching technique in a zero-sum game, then the average over the mixed strategies converges to Nash equilibrium strategies.
Assume we play rock, paper, scissors, and player 1 uses regret matching.

1. Initial cumulative regret is $(0, 0, 0)$.
2. If there is no positive accumulated regret, play uniform strategy $(1/3, 1/3, 1/3)$.
3. Player 1 chooses $R$, player 2 $P$.
4. Regret for player 1:
   - $R : u_1(R, P) - u_1(R, P) = -1 - -1 = 0$
   - $P : u_1(P, P) - u_1(R, P) = 0 - -1 = +1$
   - $S : u_1(S, P) - u_1(R, P) = 1 - -1 = +2$
5. Player 1’s cumulative regret is now $(0, 1, 2)$.
6. Regret matching suggests this strategy: $\alpha_1 = (0, 1/3, 2/3)$.
7. Player 1 chooses $P$, while player 2 chooses $S$. 
Motivation
Kuhn Poker
Real Poker: Problems and techniques
Counterfactual regret minimization

Regret matching: RPS example with two rounds II

8 Regret for player 1:
- \( R : u_1(R, S) - u_1(P, S) = 1 - -1 = +2 \)
- \( P : u_1(P, S) - u_1(P, S) = -1 - -1 = 0 \)
- \( S : u_1(S, S) - u_1(P, S) = 0 - -1 = +1 \)

9 Cumulative regret is now (2, 1, 3)

10 Regret matching: \( \alpha_1^2 = (1/3, 1/6, 1/2) \)

11 The average strategy is (1/6, 3/12, 7/12). Well, not close to the NE strategy, but will converge!
Counterfactual regret minimization

- Regret matching in strategic games does not buy us anything. We know how to compute NEs for zero-sum games already.

- In extensive-form games, we can use it to modify our behavioral strategies at each information set.

- We have to “pass down” the probability that an information set is reached and have to “pass up” the utility of a terminal history.

- As in the strategic game case, the average strategy converges to a Nash equilibrium (in behavioral strategies).

- Is it good enough?

- Since a lot of histories are explored, also “off-NE strategies” will be visited and reasonable choice will occur.
Notation & Definitions I

- During training, $t$ and $T$ denote time steps.
- Let $\pi^\beta(h)$ be the probability that history $h$ will be reached (depends on behavioral strategy profile $\beta$ and chance moves).
- $\pi^\beta(I_i) = \sum_{h \in I_i} \pi^\beta(h)$ is then the probability that information set $I_i$ will be reached.
- The counterfactual reach probability of $I_i$, written $\pi^\beta_{-i}(I_i)$, is the probability of reaching $I_i$ under the assumption that player $i$ always uses actions with probability 1 in order to reach $I_i$.
- If $\beta$ is a behavioral strategy profile, then $\beta_{l_i \rightarrow a}$ is the same profile, except that at information set $I_i$, player $i$ always plays $a$. 
Notation & Definitions II

- If $z \in Z$ is a terminal history, then we write $h \sqsubset z$, if $h$ is a proper prefix of $z$.
- For $h \sqsubset z$, the notation $\pi^\beta(h, z)$ is the probability that we reach $z$ from $h$.
- The **counterfactual utility** of $\beta$ at non-terminal history $h$ is:
  \[
  v_i(\beta, h) = \sum_{z \in Z, h \sqsubset z} \pi^\beta_{-i}(h) \pi^\beta(h, z) u_i(z).
  \]
- The **counterfactual regret** of not taking action $a$ at history $h \in I_i$ is:
  \[
  r(h, a) = v_i(\beta_{I_i \rightarrow a}, h) - v_i(\beta, h).
  \]
Counterfactual regret of not taking $a$ at $l_i$:

$$r(l, a) = \sum_{h \in l_i} r(h, a).$$

$r^t_i(l_i, a)$ refers to the regret in episode $t$, when players use $\beta^t$ and $i$ does not $a$ in $l_i$.

Cumulative counterfactual regret is then defined as:

$$R^T_i(l_i, a) = \sum_{t=1}^{T} r^t_i(l_i, a).$$

Let us define the positive cumulative counterfactual regret as:

$$R^{T,+}_i(l_i, a) = \max(R^T_i(l_i, a), 0).$$
Now, the **regret matching strategy** for episode $T + 1$ is called $\beta^{T+1}$ and computed as:

$$
\beta^{T+1}(l_i, a) = \begin{cases} 
\frac{R_{i}^{T,+}(l_i,a)}{\sum_{a \in A(l_i)} R_{i}^{T,+}(l_i,a)} & \text{if } \sum_{a \in a(l_i)} R_{i}^{T,+}(l_i,a) > 0 \\
\frac{1}{A(l_i)} & \text{otherwise.}
\end{cases}
$$
CFR in action

- One use usually what is called chance sampling, i.e., one uses one or more shuffles of the cards to compute the values for one episode.
- That also means that only a small part of the game tree needs to be in main memory.
- After a fixed number of episodes one stop and then has an (approximate) NE.
- Although, we would have liked a sequential equilibrium, we most probably will also collect regret values for information set, which are not on equilibrium profile histories.
- There are many variations and refinements of CFR.
- Looks like reinforcement learning, but it is not.