2013-Neller

An Introduction to Counterfactual Regret Minimization

- **Date**: 2013-07-09
- **Link:** Link
- **Authors:**
  - Neller, Todd
  - Lanctot, Marc
- **Cites:**
  - 2000-Hart
  - 2007-Zinkevich
  - 2009-Lanctot
- **Cited by:**
- **Keywords:** #counterfactual-regret-minimization
- **Collections:**
- **Status:** #in-progress

---

1 - Motivation

- *Regret matching* - players reach equilibrium play by tracking regrets for past plays, making future plays proportional to positive regrets.
  - First introduced by 2000-Hart.

- Goal of the paper is to provide an introduction to regret-based algorithms.

- Overview:
  - Section 2 - player regret and the regret-matching algorithm.
  - Section 3 - counterfactual regret minimization (CFR)
  - Section 4 - methods for cleaning approximate policies.
  - Section 5 - CFR with repeated states w/ imperfect recall.
  - Section 6 - Open research problems.
  - Section 7 - Further challenge problems.

2 - Regret in Games

- Rock-Paper-Scissors (RPS) will be used as an example to illustrate the regret matching algorithm.

2.1 - Rock-Paper-Scissors

- Two-player game where the players make a simultaneous action.
  - Action set: rock, paper, scissors.
  - Depending on chosen action combos, players can win, lose, or draw.

2.2 - Game Theoretic Definitions

- *Big Idea* - What does it mean to play a game optimally when maximizing wins minus losses depends on how the opponent plays?
  - Different answers to this question represent different *solution concepts*.

- *Normal form game* - tuple (N, A, u), where:
  - $N = \{1, \ldots, n\}$ - finite set of n players.
  - S_i - finite set of actions for player i.
  - $A = S_1 \times \ldots \times S_n$ - set of all possible combination of actions of all players.
    - *Action profile* - element in A - combination of player actions.
  - u - payoff function that mapping action profiles to a vector of utilities for each player.

- Because NFGs are "one-shot" games, they can be expressed as tables.
  (See the paper for the RPS table)

- *Zero-sum games* - games in which the utility vector sums to zero.

- *Pure strategy* - if the player chooses an action with probability 1.

- *Mixed strategy* (σ) - if the player has at least two actions that are played with non-zero probability.

  - Let $\sigma_i(s)$ be the probability player i selects action $s \in S_i$.

- By convention, $-i$ refers to player i's opponent.

- *Expected utility* - For the two-player case:

$$u_i(\sigma_i,\,\sigma_{-i}) = \sum_{s\in S_i}\sum_{s'\in S_{-i}} \sigma_i(s)\,\sigma_{-i}(s')\,u_i(s,s')$$

- *Best response* - strategy for player i that maximizes the expected utility for player i, given all other possible player strategies.

- *Nash equilibrium* - the combination of strategies where all the players in a game are playing their best response strategy.
  - i.e. no player can expect to improve by changing their strategy alone.
  - This is an example of a *solution concept*.

- *Correlated equilibrium* - more general solution concept that allows for players to observe a random signal from a third-party before they choose their action.
  - Players can correlate their actions to the signal, allowing for interesting solutions.
  - *Ex:* Battle of the Sexes (see paper) - player's could correlate always choosing movie or game based on a binary signal, allowing for a greater expected utility than the NE.

2.3 - Regret Matching and Minimization

- RPS setup
  - Suppose we are playing for money, each player antes a dollar.
  - Winner takes all and money back for draws.
  - The player's utility is their net money won or lost (-1, 0, +1).

- *Regret* - the difference between the utility of that action and the utility of the action we actually chose, w.r.t. the fixed choices of other players:

$$\text{regret} = u(s'_i, s_{-i}) - u(a)$$

  where $a \in A$ is the action profile, (s_i, s_{-i}), and s'_i is an action player i could've played.

  - *Ex:* if we play rock when our opponent plays paper, then we regret not playing paper for the draw (regret = 1), but really regret not playing scissors for the win (regret = 2).

  - Note - the regret for a best response action is zero.

- *Regret matching* - agent actions are selected at random with a distribution that is proportional to *positive* regrets.

  - Positive regrets indicate how much we wish we would've played that action in the past.

  - We normalize the positive regrets to get a well-formed strategy.

  - *Note* - we can't just always choose the action we regret most not having played in the past as this would be highly exploitable.

- *Cumulative regret* - regret for actions that we accumulate over multiple game iterations.

- Regret matching algorithm:

  - For each player, initialize all cumulative regrets to 0.

  - For some number of iterations:

    - Compute a regret-matching strategy profile.
      (If all regrets are non-positive, use a uniform, mixed strategy.)

    - Add the strategy profile to the strategy profile sum.

    - Select each player's action according to the strategy profile.

    - Compute player regrets.

    - Add player regrets to player cumulative regrets.

  - Return the average strategy profile.
    (i.e. strategy profile sum divided by the number of iterations.)

- The regret matching algorithm converges to a correlated equilibrium.

2.5 - Exercise: RPS Equilibrium

- See my GitHub repo - regret-matching-rpc

3 - Counterfactual Regret Minimization

- This section extends regret matching to sequential games.

3.1 - Kuhn Poker Defined

- Simple 3-card poker game with one round of betting allowed.

- Each player must ante 1 chip.

3.2 - Sequential Games and Extensive Form Representation

- *Sequential games* - games in which play consists of a sequence of actions.
  - *Note* - these games can be reformulated as a normal-form game where the players choose from a pure set of strategies at the beginning of the game.

- *Extensive form representation* - representation of sequential games as a game tree of states with edges representing transitions from state to state:

- Types of nodes in extensive-form games:
  - *Chance node* - models chance events, each edge representing a possible outcome.
  - *Decision node* - models the player, each edge represents an action available to the player.

- *Information set* - set of decision nodes for a player where all information available to that player is the same, at that decision.
  - i.e. the player can not distinguish which node she is in for that information set.

- *Partially observable* - property of games for which player's can not observe the full game state.
  - Ex. In Kuhn poker, player's can not observe their opponent's card.
  - The presence of partial observability creates information sets.

3.3 - Counterfactual Regret Minimization

- Additional considerations to CFR on top of the RM algorithm:

  1. The probability of reaching each information set given the player's strategies.

  2. Sequential nature of the game tree - game state and action sequence probabilities are passed forward, utilities are passed backward.

- CFR papers: 2007-Zinkevich and 2009-Lanctot.

- Notation:
  - A - set of all game actions.
  - I - information set.
  - $A(I)$ - set of legal actions for the information set I.
  - T - time step.
  - t - time step wrt each information set, incremented for each visit.
  - σ_i^t - strategy, assigns a probability distribution over legal actions for player i.
  - σ^t - strategy profile (i.e. set of all player strategy combinations).
  - σ_{-i} - strategy profile excluding player i.
  - $\sigma_{I \to a}$ - strategy profile where a is always chosen at I.

- *History* (h) - sequence of actions (including chance) starting at the root of the game tree.
  - $\pi^\sigma(h)$ - probability of h occurring with strategy profile σ.
  - $\pi^\sigma(I)$ - probability of reaching I through all possible game histories.
  - Z - set of terminal game histories, sequences from root to leaf.

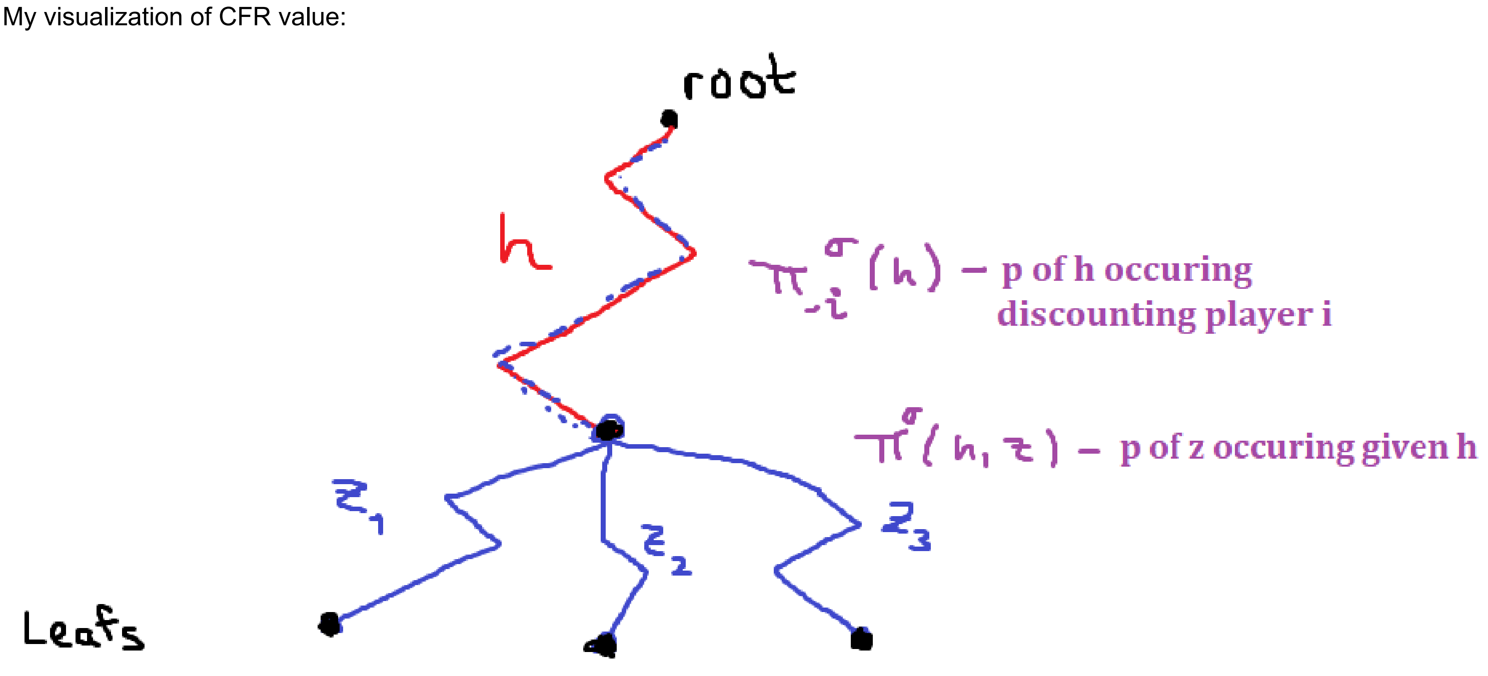
- *Counterfactual reach probability* $\pi_{-i}^\sigma(I)$ - probability of reaching I with strategy profile σ, except player i's actions to reach the state are taken as probability 1.
  - i.e. treat it as though player i is playing as though to reach I.

- *Counterfactual value* - the value at a nonterminal history h:

$$v_i(\sigma,\, h) = \sum_{z \in Z,\, h \sqsubset z} \pi_{-i}^\sigma(h)\, \pi^\sigma(h,\, z)\, u_i(z)$$

  Note: we are summing over the set of terminal histories that contain h as a subset sequence.

- My visualization of CFR value:



- *Counterfactual regret* -
  Regret of not taking action a at history h:

$$r(h,\, a) = v_i(\sigma_{I \to a},\, h) - v_i(\sigma,\, h)$$

Regret of not taking action a at information set I:

$$r(I,\ a) = \sum_{h \in I} r(h,\ a)$$

> *Key Note* - the difference between the value of always choosing action a and the expected value when the players use σ is an action's regret, which is then weighted by the probability that the other players (including chance) will play to reach the node.

- Cumulative counterfactual regret:

$$R_i^T(I,a) = \sum_{t=1}^{T} r_i^t(I,a)$$