Estimating Poker Hand Strength Using MCMC

A probabilistic approach to poker strategy through Markov Chain Monte Carlo simulation.

By Adam Amar

Presentation Overview

Poker Fundamentals

Rules and key definitions

Project Motivation

The challenge of hidden information

MCMC Methodology

Formulas and implementation details

Results & Applications

Simulation outcomes and insights

Poker 101: Essential Concepts

Game Structure

Simplified Texas Hold'em with 6 players, post-Turn stage.

Card Distribution

Each player gets 2 private hole cards. Five community cards shared.

Winning Conditions

Best 5-card hand wins at showdown. Strategic betting also matters.


WHAT'S FLOP, TURN, AND THE RIVER IN POKER?
FLOP
TURN
RIVER


adda52.com
POKER HANDS
1. ROYAL FLUSH
2. STRAIGHT FLUSH
3. FOUR OF A KIND
4. FULL HOUSE
5. FLUSH
6. STRAIGHT
7. THREE OF A KIND
8. TWO PAIR
9. PAIR
10. HIGH CARD

The Core Challenge

What We Know

- Our two hole cards
- Four community cards
- Betting patterns

What We Don't Know

- Opponents' hole cards
- Final community card
- True hand strength

Our decision quality depends on estimating relative hand strength against plausible opponent holdings.

Why Poker Hand Strength Matters

Context Dependency

The same hand can be strong or weak depending on the situation. AK suited is strong preflop but weak on a 2-7-9 board.

Theoretical Applications

Perfect case study for probability modeling with hidden information. Demonstrates Bayesian inference in action.

Practical Value

Accurate hand strength estimation directly translates to better decision-making and profit in real games.

MCMC as the Ideal Solution

Sampling Complexity

MCMC excels at sampling from complex probability distributions.

Likelihood Filtering

Favors plausible opponent hands through likelihood function.

Convergence Properties

Provides theoretical guarantees on estimate quality.

Dimensional Reduction

Makes an enormous search space tractable.

Project Objective

Model opponents

Assign plausible hands to four opponents

Categorize hands

Evaluate using standard poker categories

Compute likelihoods

Determine probability of each configuration

Refine via MCMC

Iteratively improve estimates to obtain final %

Modeling Opponent Behavior

Factor	Adjustment	Rationale
Base Winrate	By hand category	Intrinsic strength
Aggressive Play	+0.07	Signals confidence
Passive Play	-0.05	Signals weakness
Position Advantage	+0.03	Last to act

Winrate formula:

$$w_i = \min(0.99, \max(0.01, \text{base}(H_i) + \Delta_{\text{aggression}} + \Delta_{\text{position}}))$$

MCMC Simulation Logic

Initialize Random Table

Begin with a random distribution of opponent hands.

Compute Total Likelihood

Calculate

$$L = \prod_{i=1}^{n} w_i$$

as the product of all winrates.

Propose New Configuration

Generate alternative opponent hand assignments.

Accept or Reject

Use probability:

$$P(accept) = \min\left(1, \frac{L_{new}}{L_{current}}\right)$$

Track Convergence

Monitor likelihood trace over simulation iterations.

Implementation Details

Technology Stack

Python with Treys library for hand evaluation

Core Functions

- generate_mcmc_input()
- adjusted_winrate()
- mcmc_simulation()

Analysis Output

Trace of likelihoods and hand distributions

Simulation Parameters

1,000 iterations with 100 burn-in period

Overcoming Implementation Challenges

Treys Card Mismatch

Problem: Inconsistent card format representations

Solution: Reordered and lowercased suit/rank identifiers

Missing Hole Cards

Problem: PHH file lacked necessary card data

Solution: Switched to pure simulation approach

MCMC Instability

Problem: Erratic convergence behavior

Solution: Implemented burn-in period and trace logging

Unrealistic Behavior

Problem: Model didn't reflect real poker strategy

Solution: Added aggression and position adjustments

MCMC Likelihood Trace Analysis

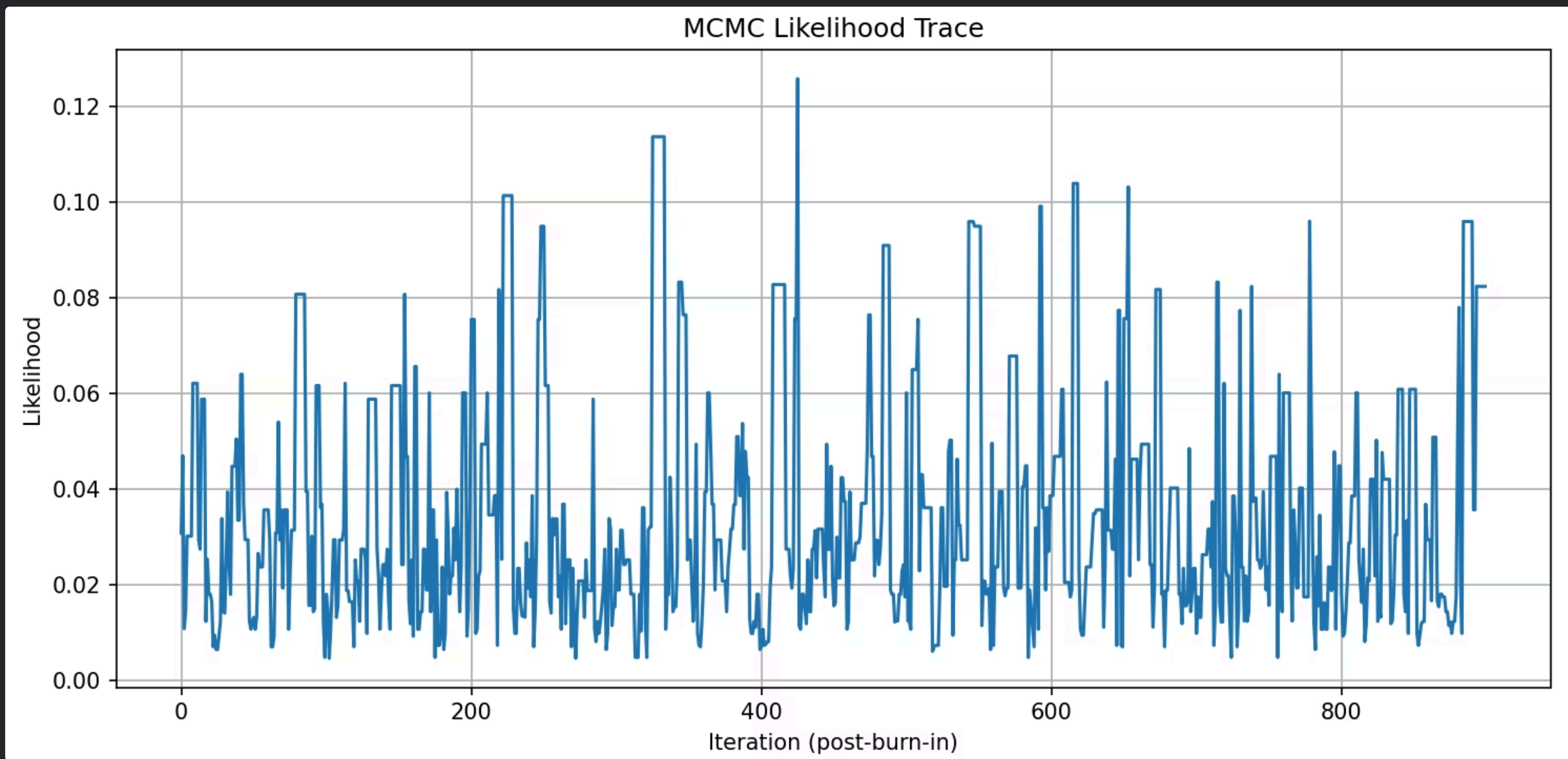
Key Metrics

- Burn-in period: 100 iterations
- Acceptance rate: ~30%
- Stabilization: After ~500 iterations

Trace Interpretation

Fluctuations show the sampler exploring the configuration space. Higher values indicate more plausible hand combinations.

Occasional drops represent strategic exploration of suboptimal states, essential for thorough sampling.



Sample MCMC Output

Your hand: High Card [HA, DK] | WinRate: 0.18

Board: [C9, D7, S8, HQ]

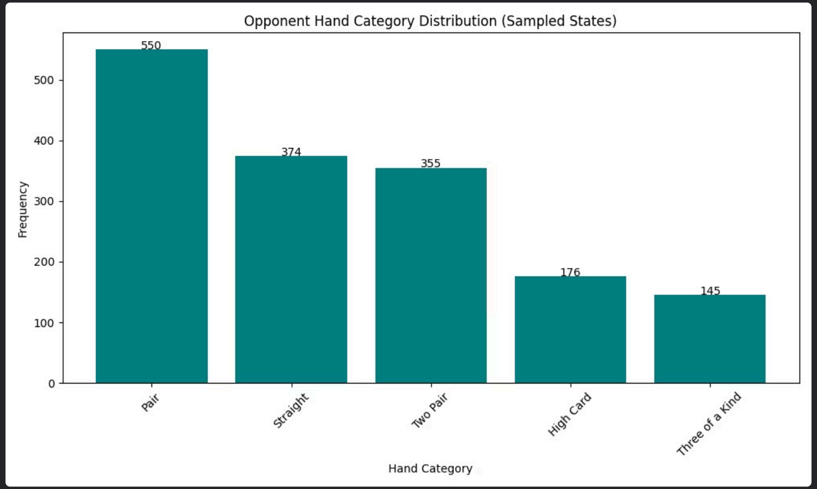
Player	Cards	Category	WinRate
Opponent 1	[SK, D8]	Pair	0.35
Opponent 2	[C2, D4]	High Card	0.03
Opponent 3	[DT, DJ]	Straight	0.85
Opponent 4	[D2, C6]	High Card	0.18

Key Results and Interpretation

Our MCMC simulation reached convergence successfully with a healthy 30% acceptance rate, indicating efficient exploration of the probability space.

The predicted opponent hand distribution reflected realistic poker scenarios, with three moderate hands and one dominant straight.

Our Ace-King high card showed only 18% win probability - dramatically lower than other estimates would suggest.



Future Extensions

Future Extensions

Adding River analysis, multi-player equity, and opponent modeling improvements.

Our next development phases will focus on enhancing the algorithm's capabilities and expanding its application scope.

Questions?