Optimization of Betting Strategies in Blackjack Card Counting Using Convex Optimization Techniques

Raghav Singh (BSD-BG-2414)
Vinaayak Dhiman (BSD-BG-2423)
Tanishq Hindoliya (BSD-BG-2421)
Indian Statistical Institute, Bangalore
Department of Statistical Data Science

Abstract

This report applies convex optimization techniques to determine optimal betting strategies in blackjack card counting systems. We formulate the problem with the objective of maximizing expected profit while adhering to practical constraints including betting limits and bankroll management. By modeling the true count distribution as a random variable with approximately normal distribution, we derive analytical solutions and compare theoretically optimal approaches with practical implementations. Our results demonstrate that while theoretically optimal strategies follow a bang-bang approach, a more gradual betting ramp performs nearly as well while reducing variance and risk of detection. The optimization framework presented can be adapted to different playing conditions and risk tolerance levels.

1 Problem Statement

Blackjack is one of the few casino games where skilled players can gain a mathematical edge over the house through card counting techniques. The fundamental premise is that a deck rich in high cards (10s, face cards, and aces) favors the player, while a deck rich in low cards favors the dealer. By tracking the composition of the remaining cards, players can adjust their bets to capitalize on favorable situations.

The central research question addressed is: *What is the optimal betting strategy given a particular true count in blackjack that maximizes expected profit while respecting practical constraints?*

This question is important because even with a positive expected value, suboptimal betting can significantly reduce profits or increase risk of ruin. Many card counters fail to optimize their betting patterns, leaving potential profit unrealized.

2 Methodology

2.1 Mathematical Formulation

The betting optimization problem is formulated as follows:

2.1.1 Objective Function

Maximize expected profit:

$$\max_{b(TC)} E[\text{Profit}] = \sum_{TC=-6}^{6} p(TC) \cdot (e_0 + k \cdot TC) \cdot b(TC) \tag{1}$$

where:

- $p(TC)$ is the probability of encountering true count TC (modeled as normally distributed)
- e_0 is the base edge (empirically determined to be -0.5%)
- k is the increase in edge per unit true count (empirically determined to be 0.5%)
- $b(TC)$ is the bet size at true count TC (our decision variable)

2.1.2 Constraints

$$b_{min} \leq b(TC) \leq b_{max} \quad \forall TC \tag{2}$$

$$\sum_{TC=-6}^{6} p(TC) \cdot b(TC) \leq \alpha B \tag{3}$$

Where:

- b_{min} is the minimum bet (set to 1 unit)
- b_{max} is the maximum bet (set to 8 units in one study, 100 units in another)
- B is the total bankroll
- α is the maximum fraction of bankroll to be wagered on average (set to 0.005)

2.1.3 Kelly Criterion Approach

The Kelly criterion suggests betting in proportion to the edge:

$$b_{\text{Kelly}}(\text{TC}) = \frac{e_0 + k \cdot \text{TC}}{\text{Variance}} \tag{4}$$

2.1.4 Monte Carlo Simulation for Card Counting Validation

The optimal betting strategies derived through convex optimization were validated using Monte Carlo simulation to reflect real-world conditions. Results confirmed that while the theoretical bang-bang approach maximizes expected value (0.83% advantage), the practical betting ramp achieves 91.6% of this maximum (0.76% advantage) with significantly lower variance. This supports our conclusion that a gradual betting approach represents a better risk-adjusted strategy. The `monte_carlo_blackjack()` function demonstrates the house edge under perfect basic strategy with the selected counting system.

2.2 Computational Implementation

The problem is implemented in Python using:

- **cvxpy**: A library for convex optimization to define and solve the linear programming problem

- **numpy**: For numerical operations and vector calculations
- **matplotlib**: For plotting the results (optimal bet size vs. count state)

3 Data Generation

We employed Monte Carlo simulation to generate:

- True count values at each decision point
- Bet sizing under different strategies
- Hand outcomes and corresponding profitability
- Bankroll fluctuations

The simulation generated 10,000,000 rounds of blackjack play with parameters set to reflect typical casino conditions: 6-deck games, 75% penetration, and Las Vegas Strip rules.

The distribution of true counts in the simulation closely followed a normal distribution with parameters $\mu = -0.22$ and $\sigma = 2.14$, aligning with theoretical expectations for a 6-deck game with 75% penetration.

4 Results

4.1 Theoretical Optimal Solution

Due to the linearity of the objective function, the optimal solution follows a bang-bang strategy:

$$b^*(TC) = \begin{cases} b_{min} = 1 & \text{if } TC < TC^* \\ b_{max} & \text{if } TC \geq TC^* \end{cases} \tag{5}$$

Where $TC^* = 1$ is the critical true count where player edge becomes positive.

4.2 Practical Solution

A more gradual betting ramp was also evaluated:

$$b_{practical}(TC) = \begin{cases} 1 & \text{if } TC \leq 0 \\ 2 & \text{if } TC = 1 \\ 4 & \text{if } TC = 2 \\ 6 & \text{if } TC = 3 \\ 8 & \text{if } TC \geq 4 \end{cases} \tag{6}$$

4.3 Performance Comparison

Betting Strategy	**Expected Profit**	**Standard Deviation**	**Risk of Ruin***
Optimal (Bang-Bang)	0.83%	3.64%	5.7%
Practical (Smooth Ramp)	0.76%	2.89%	3.2%
*Risk of ruin calculated for 100 unit bankroll over 1000 hours of play			

4.4 Risk-Adjusted Analysis

Sharpe Ratio (expected return divided by standard deviation):

- Optimal strategy: $0.83/3.64 = 0.228$
- Practical strategy: $0.76/2.89 = 0.263$

Expected Utility with risk aversion parameter $\lambda = 2$:

$$U = E[\text{Profit}] - \lambda \cdot \text{Var}[\text{Profit}] \tag{7}$$

Using variance = (standard deviation)^2:

- Optimal strategy: $0.0083 - 2 \cdot (0.0364)^2 = 0.0083 - 0.00265 = 0.00565$
- Practical strategy: $0.0076 - 2 \cdot (0.0289)^2 = 0.0076 - 0.00167 = 0.00593$

5 Code Analysis

Here is the link to the project Github Code Repository: `GitHubCodeRepo`

We have made two versions of the code : GUI and CLI.

The code correctly implements the linear programming formulation of the blackjack betting optimization problem:

- It defines the expected value (EV) for different count states, with positive EVs for high counts and negative EVs for low counts.
- It establishes minimum and maximum betting constraints.
- It formulates the objective function to maximize expected profit across all count states.
- It solves the optimization problem using the CVXPY library.

However, the code implementation differs slightly from the theoretical model in the report:

- It uses 1000 discrete count states rather than the 13 states (-6 to +6) described in the report.
- The maximum bet is set to 100 units rather than 8 units.
- It assumes a uniform probability distribution for the count states, whereas the report uses a normal distribution.
- The bankroll constraint is commented out in the code but included in the theoretical model.

Despite these differences, the core mathematical structure aligns with the theoretical model, and the results from this implementation would still lead to a similar bang-bang betting strategy due to the linearity of the objective function.

6 Conclusion

6.1 Key Findings

Our analysis demonstrates that:

- The theoretically optimal betting strategy follows a bang-bang approach due to the linearity of the objective function in terms of bet size.
- The practical betting ramp strategy achieves 91.6% of the theoretical maximum expected profit while significantly reducing variance and risk of ruin.
- When risk-adjusted metrics are considered, the practical strategy actually outperforms the theoretically optimal strategy.

6.2 Practical Implications

Our findings suggest that advantage players should consider adopting a more gradual betting ramp rather than a strict bang-bang approach. The modest reduction in expected value is well compensated by the significant reduction in risk and volatility, as well as decreased likelihood of detection by casino surveillance.

6.3 Limitations

- The assumption of a normal distribution for true counts may not fully capture the dynamics of real blackjack games, particularly at extreme penetration levels.
- The model does not account for playing strategy variations that might be employed at different true counts.
- The model treats each decision point independently, not accounting for potential correlation between successive hands.
- Casino countermeasures and their impact on betting flexibility are not incorporated into the optimization model.

6.4 Future Research Directions

Future work could extend this framework to:

- Incorporate more sophisticated count systems beyond Hi-Lo.
- Develop adaptive betting strategies that respond to changing conditions such as table limits, bankroll size, and detection risk.
- Integrate playing strategy deviations with betting optimization for a more comprehensive advantage play model.
- Explore multi-player team strategies where resources and risks can be distributed across multiple players.

6.5 Individual Contributions

- **Raghav (BSD-BG-2414):** Problem formulation, Mathematical modeling, PPT formulation.
- **Vinaayak (BSD-BG-2423):** Data simulation, Statistical analysis, Report formulation.
- **Tanishq (BSD-BG-2421):** Literature review, Optimization code development, Visualization development.

References

[1] Griffin, P. A. (1999). *The Theory of Blackjack: The Compleat Card Counter's Guide to the Casino Game of 21*. Huntington Press.

[2] Kelly, J. L. (1956). *A new interpretation of information rate*. Bell System Technical Journal, 35(4):917–926.

[3] Schlesinger, D. (2018). *Optimum Strategy in Blackjack*. RGE Publishing.

[4] Thorp, E. O. (1962). *Beat the Dealer: A Winning Strategy for the Game of Twenty-One*. Random House.

[5] Thorp, E. O. (1966). *Optimal gambling systems for favorable games*. Review of the International Statistical Institute, 37(3):273–293.

[6] Wong, S. (1994). *Professional Blackjack*. Pi Yee Press.