Ph.D. Tonny Mayambala
tonnymayambala.tm@gmail.com
Saint Petersburg | Kampala
+79992056874 | +256760705065

Poker Analysis

♠ ♥ ♣ ♦

Table of Contents

1. Task

Design a model that derives interesting insights from the datasets and present the insights that has statistical evidence to ensure the results are good. You are allowed to use any libraries in python that you need no restriction. Code should be in python 3.12 and higher versions. Code should be properly commented and should be clean.

Poker Dataset link : https://poker.cs.ualberta.ca/IRC/IRCdata.tgz
Unstable Python Library for the dataset:
https://github.com/allenfrostline/PokerHandsDataset
Libraries you might need

- https://pytorch.org/
- https://scipy.org/
- https://numpy.org/
- https://pandas.pydata.org/

Techniques you are expected to use Machine Learning or Neural Networks. Potentially things like Boosting algorithms or any other things that might be of interest.

Deliverables: You are expected to send the code that was used for analysis, ensuring the code runs properly. Provide a README.md file which has directions of how to run the code. Provide a pdf document that explains the insights in detail and assume your audience are a non-technical audience with only a Bachelor's level of education and most likely do not have a strong math background.

2. Introduction

Having closely analysed Michael Maurer's insights on **Short-Handed Holdem Variance** and **Variance for 3-6 hold'em**, I started off with analysing research of the brains behind the IRC database. All 25 past members including outstanding research from *Jonathan Schaeffer*, and his book "*man vs machine*",, definitely caught my eye. General insights on game theory manipulations from different IRC forums and their current 4 members with exception to ***Dr. Michael Bowling*** [1], from the University of Alberta and *Reinforcement Learning group*.

Financial Markets are my strong hand and are closely intertwined with Pokermetrics, Game theory, machine learning and Artificial intelligence! I tried to focus on the risk management side, while doing the task, however I must highlight that a lot has to do with the behavioral and psychological elements [2], whose data was not available for example; Replicator Dy-namics (RD- how players switch between different strategies), meta strategies (behaviour over a series of games), stability etc.. [3].

Understanding both statistical insight examples on how the task was performed drew my attention to risk management immediately having lost historically, during my poker losing streaks. I replicated the insights but added an "*aggression factor*" to understand how that affected results. With this being done, I could test out different strategies to assess how P&L would be affected. Probability theory and strategies like, martingale [9, 10] , kelly formula and others seemed a little lengthy but probable in the future.

Finishing up with a predictive light GBM model calibrated by statistical *performance strength* over hands used and aggression factors to enhance the training process.

3. Statistical analysis

The top 15 players in this table have demonstrated strong performance in their poker games. (see pic. 1)

	player_name	hands_count	hands_won_count	win_aggression_factor	loss_aggression_factor	actions	profit	return	SB_per_hand	bankroll_mean	bankroll_std
0	sagerbot	68656	22548	1.666	1.666	7801100.0	359978.0	1.046	5.243	186187.009	166275.786
1	DrOakland	48931	17040	2.345	2.345	4798765.0	153150.0	1.032	3.130	59597.840	40736.431
2	Benway	30464	12242	1.486	1.486	3738325.0	146462.0	1.039	4.808	44615.791	31292.251
3	superman	12949	5092	1.965	1.965	1490900.0	89725.0	1.060	6.929	214121.010	263069.667
4	mt	17961	6630	1.934	1.934	1869047.0	87622.0	1.047	4.878	225516.830	166833.240
5	exosis	12846	5061	1.669	1.669	1813800.0	83050.0	1.046	6.465	76035.976	58007.700
6	Gunshot	8867	3796	2.392	2.392	1414861.0	81606.0	1.058	9.203	25747.110	16924.952
7	Blades	1493	488	1.479	1.479	224093.0	77589.0	1.346	51.969	29251.285	22444.816
8	po147	10358	3764	1.056	1.056	1434442.0	70236.0	1.049	6.781	31774.834	19277.257
9	KisMyAce	13559	4434	1.090	1.090	1638785.0	68685.0	1.042	5.066	82802.137	31933.582
10	jvegas2	19620	7039	1.358	1.358	2737039.0	60323.0	1.022	3.075	101442.590	70116.421
11	DM	8107	3264	1.278	1.278	1220142.0	55008.0	1.045	6.785	53864.707	40309.431
12	gazoo	3746	1472	1.734	1.734	731425.0	51875.0	1.071	13.848	42331.658	34386.886
13	show	10959	4607	1.162	1.162	1514800.0	48675.0	1.032	4.442	49224.428	30634.373
14	GOD	821	248	2.089	2.089	138075.0	47175.0	1.342	57.460	402581.326	180644.819

Pic 1. The top 15 players

They have played a high number of hands, with players like “sagerbot” leading with 68,656 hands. Their aggression factors, both in winning and losing hands, indicate a consistent, strategic approach to betting and raising, with “DrOakland” showing the highest win and loss aggression factors at 2.345. The actions taken by these players, such as bets, raises, and folds, are substantial, with the highest being 7,801,100 by “sagerbot.” Profit values indicate their success, with positive profits like 359,978 for “sagerbot.”

The return column shows a favorable return on investment, with values above 1.00 for most players, signifying profitable gameplay. The SB_per_hand (small bets per hand) values and bankroll metrics further highlight their effective financial management, with “GOD” having the highest bankroll_std of 180,644.819, indicating some fluctuation but overall positive outcomes.

In contrast, the bottom 15 players have shown less favorable performance metrics. (see pic. 2 below)

	player_name	hands_count	hands_won_count	win_aggression_factor	loss_aggression_factor	actions	profit	return	SB_per_hand	bankroll_mean	bankroll_std
0	VWVW	3682	1079	1.270	1.270	337975.0	-15450.0	0.954	-4.196	7462.241	3475.983
1	Beelzebub	1231	347	1.674	1.674	196925.0	-15700.0	0.920	-12.754	77474.034	20823.560
2	XTC	14	4	48.500	48.500	18655.0	-17505.0	0.062	-1250.357	65578.357	6793.113
3	PoTz	2180	783	2.003	2.003	236008.0	-17983.0	0.924	-8.249	10001.232	4806.434
4	sabylbot	4247	1046	1.171	1.171	477325.0	-18675.0	0.961	-4.397	103886.105	87998.113
5	snap	5	3	21.000	21.000	20350.0	-19975.0	0.018	-3995.000	25935.000	8940.889
6	fuhzee	6790	1181	0.642	0.642	494850.0	-22600.0	0.954	-3.328	211160.619	233382.790
7	JackHoff	45	8	3.607	3.607	27543.0	-23711.0	0.139	-526.911	3392.600	3096.996
8	K6	6926	2125	1.218	1.218	656925.0	-25225.0	0.962	-3.642	361841.343	217538.984
9	jpsych	2374	855	0.982	0.982	408675.0	-26325.0	0.936	-11.089	134899.695	64929.623
10	BlackBart	26594	6521	0.692	0.692	2271763.0	-33515.0	0.985	-1.260	248562.286	259703.842
11	r00lbot	10899	895	1.087	1.087	503100.0	-35100.0	0.930	-3.220	85192.991	40970.294
12	Cuda	93	26	3.014	3.014	44586.0	-35361.0	0.207	-380.226	37841.882	5247.158
13	Yolanda	61	0	0.000	4.060	49232.0	-49232.0	0.000	-807.082	14122.787	11672.357
14	Iceman	2819	835	2.050	2.050	301725.0	-54975.0	0.818	-19.502	202010.843	143022.325

Pic 2. Bottom 15 Players

Although, logically they have participated in numerous hands, their aggression factors and actions indicate different strategic approaches, with "XTC" having an exceptionally high win and loss aggression factor at 48.500, suggesting a highly *volatile* aggressive playstyle. Their profit values are predominantly negative, such as -35,100 for "*r00lbot*," highlighting consistent losses. The return on investment is below *1.00* for most, reflecting unprofitable gameplay. The SB_per_hand values indicate lower betting averages, and the bankroll metrics reveal less stable financial management. "Iceman," for instance, has a significant bankroll_std of 143,022.353, showing considerable variability in their funds, which could be linked to their aggressive yet unsuccessful strategies.

Overall, the top 15 players exhibit more disciplined and profitable strategies, while the bottom 15 players struggle with consistency and effective bankroll management.

4. Modelling.

The LightGBM model was trained to predict binary outcomes using a dataset with *64,511* data points and *7* features.

a) player_bankroll,
b) opponents_bankroll_avg,
c) actions_number,
d) player_aggression_factor,
e) opponents_aggression_factor,
f) player_hand_rank,
g) opponents_hands_ranks_avg,

```
[LightGBM] [Info] Number of positive: 33284, number of negative: 31227
[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of testing was 0.000639 seconds.
You can set `force_row_wise=true` to remove the overhead.
And if memory is not enough, you can set `force_col_wise=true`.
[LightGBM] [Info] Total Bins 1175
[LightGBM] [Info] Number of data points in the train set: 64511, number of used features: 7
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.515943 -> initscore=0.063794
[LightGBM] [Info] Start training from score 0.063794
Training until validation scores don't improve for 10 rounds
Early stopping, best iteration is:
[61]    train's binary_error: 0.178311  test's binary_error: 0.18178
Accuracy: 81.8220

Classification Report:
              precision    recall  f1-score   support

           0       0.84      0.77      0.80      3454
           1       0.80      0.86      0.83      3714

    accuracy                           0.82      7168
   macro avg       0.82      0.82      0.82      7168
weighted avg       0.82      0.82      0.82      7168
```

Pic 3. LightGBM Model

The training and testing process was efficient, using row-wise multi-threading to optimize performance. The model's binary error rates were similar for both the training and testing sets, around 0.178 and 0.182, respectively.

In terms of performance, the model achieved an overall accuracy of ***81.82%,*** meaning it correctly predicted the class in about 81.82% of the cases. For class 0, the *precision* was 0.84, indicating the proportion of correct positive predictions, while the recall was 0.77, showing the model's ability to identify all relevant instances. The f1-score, which balances precision and recall, was 0.80 for class 0.

For class 1, the precision was slightly lower at 0.80, but the recall was higher at 0.86, leading to an f1-score of 0.83. The overall precision, recall, and f1-score averages, weighted by the number of true instances for each class, were all around 0.82.

The LightGBM model performs reasonably well, especially in identifying class 1 instances, but there is room for improvement in class 0 recall. The use of optimizations like row-wise multithreading helped enhance the training process.

5. Appendix;

Aggression Factor: A measure of how often a player bets or raises compared to calling.
Performance Strength: Indicates how well a player or system is performing, often referring to the ability to make successful decisions in a game or task.
Losing Hands: Poker hands or situations that result in losing bets or pots, typically referring to hands that were outplayed.
Raising: In poker, raising is the act of increasing the current bet, forcing other players to match or raise further.
Bets: Placing a wager.
Raises: Increasing the current bet.
Folds: Surrendering the hand without betting.
Return on Investment (ROI): A percentage that measures the profit or loss made on an investment relative to the cost of that investment.
Bankroll: The total amount of money a player has set aside for gambling or investing. It's important for managing risks.
Row-wise Multi-threading: A technique in computing where multiple threads process different rows of data simultaneously to improve performance.
Precision: In data analysis, precision measures how often the positive predictions made by a model are actually correct.
F1-Score: A balance between precision and recall, used in classification models. It's the harmonic mean of the two, giving a single metric for model performance.
Kelly Formula: A mathematical formula used to determine the optimal size of a bet or investment based on the probability of winning and the odds. It helps maximize the growth of a bankroll over time.
LightGBM model: A type of gradient boosting algorithm that efficiently handles large datasets and complex patterns to predict outcomes, such as player actions or hand strength.
Precision: The ratio of true positive predictions to the total number of positive predictions (true positives plus false positives).
Recall (also known as sensitivity) is the ratio of true positive predictions to the total number of actual positive cases (true positives plus false negatives).

6. References.

1. Heads-up Limit Hold'em Poker is Solved // https://webdocs.cs.ualberta.ca/~bowling/papers/15science.pdf

2. Bayes' Bluff: Opponent Modelling in Poker // https://webdocs.cs.ualberta.ca/~bowling/papers/05uai.pdf

3. The dynamics of human behaviour in poker // https://www.researchgate.net/publication/251548824_The_dynamics_of_human_behaviour_in_poker

4. Do poker players know how good they are? Accuracy of poker skill estimation in online and offline players //https://www.sciencedirect.com/science/article/abs/pii/S0747563213004275

5. Big Two and n-Card Poker Probabilities // https://www.mdpi.com/2073-4336/14/6/73

6. Research on the Ways to Win in a Simple Poker Game // https://drpress.org/ojs/index.php/HSET/article/view/23114

7. Instruction-Driven Game Engine: A Poker Case Study // https://www.qeios.com/read/ARDYD9

8. PokAR: Facilitating Poker Play Through Augmented Reality // https://arxiv.org/pdf/2301.00505

9. Factors in Learning Dynamics Influencing Relative Strengths of Strategies in Poker Simulation // https://www.researchgate.net/publication/376064062_Factors_in_Learning_Dynamics_Influencing_Relative_Strengths_of_Strategies_in_Poker_Simulation

10. Martingales // https://link.springer.com/chapter/10.1007/978-3-031-49306-5_8

11. Asymptotics for parametric martingale posteriors // https://arxiv.org/pdf/2410.17692