Daniel Poston
February 22nd, 2018

STATISTICAL MODELING +2

Analyzing Poker Hands with Python

Analyze poker hands with Python and easily implement statistical concepts such as combinations, permutations, (in)dependent events and expected value.

In the first part of this tutorial, you learned about several concepts including Independent and Dependent events, Permutations and Combinations, Multiple events, Expected Values, and how to calculate each of them. Now you will apply those concepts to actual poker hands that I played during my career. Along the way, you'll learn much more about the concept of Expected Value and how to use it for making optimal decisions.

- First, you'll go through a short introduction to No Limit Texas Hold'em so that you have all the domain knowledge you need to complete the tutorial;

- Next, you'll also get introduced to some tools that will be used in the poker hands case studies.

- Once you have gone through these two introductory sections, you'll take a closer look at a concept that you saw in the first tutuorial: Expected Value.

- Lastly, you'll go through poker hands: Pocket 10s versus Pre Flop Raise, Pocket 8s versus Pre flop Raise, and Flop Check Raise Semi Bluff.
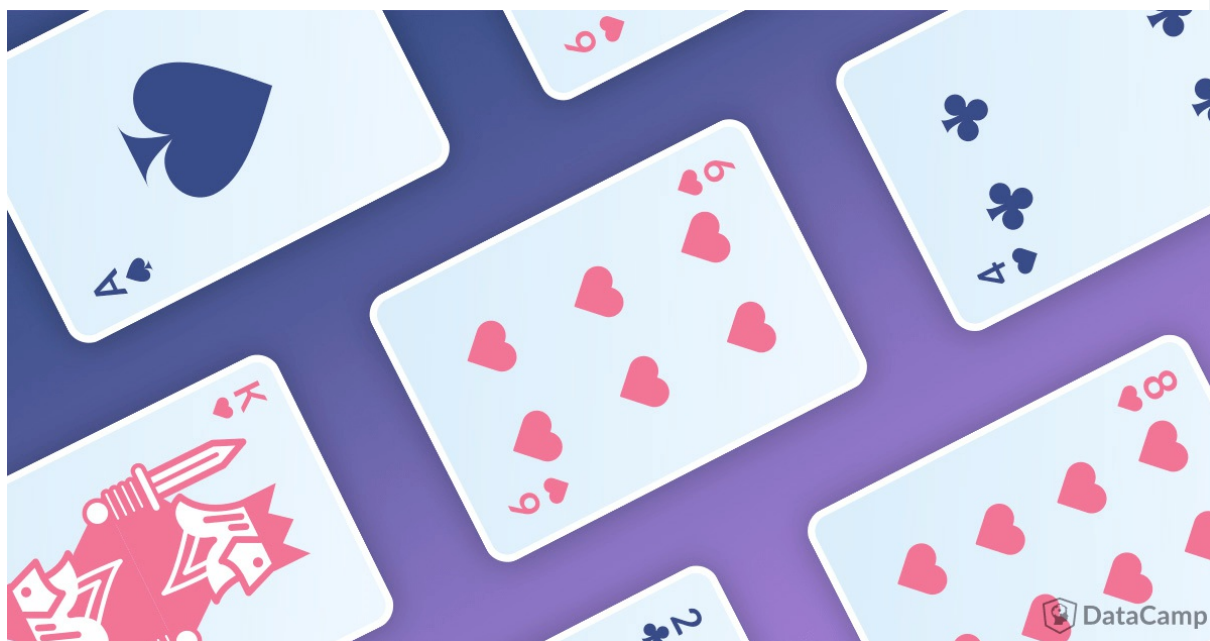
An Introduction to No Limit Texas Hold'em

Before you get started, a little background info is in order. The game that will be covered in the case studies of this article is No Limit Texas Hold'em. Played with a standard 52 card deck, Texas Hold'em is the most popular of all the poker variations. Each player tries to make the best five-card hand possible by combining their two cards with the five community cards dealt throughout the hand.

The player to the left of the dealer button starts the action and play moves clockwise around the table. The betting is typically started when the player to the left of the dealer button posts a small blind, and the next player to the left posts a big blind. Betting action takes place on multiple streets: preflop, flop, turn, and the river.

Here are a few terms you should understand:

- **Button** - The Button is a circular disk that gets passed around the table clockwise after each hand. In a home game, the person who has the Button deals the hand dealing the first card to the player on their left. At a casino, the Button identifies who is supposed to be the dealer, but the actual dealer is provided by the casino.

- **The Blinds** - The Blinds are forced bets by the two players to the left of the Button. They are called "Blinds" because the two players have to bet blind, in that they have to bet before they get to see their cards.

- **Streets** - Each round that cards are dealt is referred to as a street.

- **Pre-Flop** - This is the first betting round. Each player is dealt two cards, and the blinds start the betting action. No community cards have been dealt yet.

- **Flop** - This is the second betting round. In this round, three community cards are dealt face up in the middle of the table, followed by a round of betting.

- **Turn** - In this round, only one community card is dealt, followed by the third round of betting.

- **The River** - The final community card is dealt, followed by the last round of betting.

- **Showdown** - After the River is dealt and all betting action has taken place, any remaining players Show their cards to see who has the best hand.

- **Check** - If it's a player's turn to act, and no previous bets have been made during that round of betting, the player can Check which means to choose not to bet, and allow the next player to take their turn.

- **Bet or Raise** - Betting or Raising means to increase the size of the wager during a betting round. The first player to put money in the pot each round is making a Bet. If another player increases the size of the bet that round, they are making a Raise.

- **Call** - If it's a player's turn to act and a bet has already been made during that round, the player can match the previous bet which is known as Calling the bet.

- **Fold** -If it's a player's turn to act and a bet has already been made during that round, the player can give up and Fold their hand. This player is no longer involved in the hand and losses any money they have put in the pot.

If you've never played or are unfamiliar with poker terminology you can click here for a concise but detailed explanation.

Poker Probability Tools

Anytime I ran into a situation while playing that confused me, I would mark the hand for review later. After my poker playing session was done, I'd go back through the hands that I'd marked for review. I'd break them down mathematically so I'd have a better idea of what to do in similar situations in the future.

The hands used in this tutorial were downloaded and stored in a PostgreSQL database by Hold'em Manager, software mentioned in Part I. Hold'em Manager calculates statistics from these hand histories for every opponent you play against. The data updates and displays in real time as you play.

I picked out three hands from my poker career. You'll use statistics on my opponents that I had available at the time, deductive reasoning, and a tool called an equity calculator, to make assumptions about the probability of winning the hand for each viable option.

Note: an equity calculator is used to determine the probability of winning a hand based on a situation that you indicate. You enter your cards, the community cards (if any), and the range of hands you assume your opponent is holding. The calculator provides a probability of winning the hand if the remaining cards are dealt.

Expected Value: A Closer Look

Before getting into the analysis for the first hand, you need a better understanding of Expected Value. In Part I, you learned how to calculate your expected value for winning a \$100 dollar pot with a 75% chance of winning. There were only two possibilities:

- Win \$100

- Win \$0

With the help of the formula `ExpectedValue=TotalValue×Probability`, you calculated that the Expected Value would be \$100 * .75 or \$75.

However, many times you will not have such a straightforward calculation and you will have more than two possible outcomes.

To determine expected value, you must find the weighted average of each possible outcome, where each possible outcome is weighted by its respective probability of occuring.

That's why the equation above would be better represented like this:

$Expected Value = (100 * .75) + (0 * .25)$

$Expected Value = \$75$

And what if there were a third possible outcome for the above situation?

Take a look at the following situation, where you have a 50 percent chance of winning \$100, 25% chances of winning \$25 and another 25% chances of winning absolutely nothing:

- Win \$100 (50%)

- Win \$25 (25%)

- Win \$0 (25%)

Then, the calculation of your Expected Value will be as follows:

$Expected Value = (100 * .50) + (25 * .25) + (0 * .25)$

$Expected Value = \$56.25$

Now it's time to apply what you've learned to a few scenarios I ran into at the poker table.

Pocket 10s versus Pre Flop Raise

Note: When reviewing poker hands, it is common to refer to your opponent as Villain and the player being reviewed is referred to as Hero. The player you will be reviewing is 'GamingFoSho' which was my username at the poker sites that I played at.

Hand 1 Overview

Hero (GamingFoSho) has a pair of 10s and bet \$0.85. He has \$20.95 remaining. The Opponent or Villain raises to \$2.50 and has \$30.35 remaining.

In poker terminology, this is called a 3bet. The small blind and big blind make the first bet, and Hero raised them which was the second bet. There is currently \$3.70 total in the Pot. Calling is not a good option for reasons that are beyond the scope of this

blog post to fully explain.

Note: that, generally speaking, playing poker passively is not profitable and playing poker aggressively is profitable. Calling is passive and betting or raising is aggressive. There are other factors related to poker strategy such as player position that indicate calling is a bad option, but would require significant explanation.

Hero must decide between raising with the plan of going All-In and betting all remaining chips or folding. Folding costs nothing so you will analyze the expected value of going all in. In this situation, I'd make a small raise to induce Villain to All-in bluff, but the calculation must be done as if I'm going All-in since that is the plan.

Hero risks \$20.95 if All-in and loses. Going 'All-in' means to put all your money in the pot. Hero has \$20.95 remaining, so going All-in risks \$20.95. Hero wins \$23 if All-in and wins. There is currently \$3.70 in the pot. Hero bet \$0.85 and Villain raised to \$2.50. Hero must add \$1.65 of remaining \$20.95 to match Villain's raise, leaving Hero with \$19.30. This means Hero can win an additional \$19.30 on top of the \$3.70 already in the pot for a total of \$23.00. Since there is currently \$3.70 total in the pot, Hero wins \$3.70 if Villain folds.

```
Hand1_AllIn_Loses = -20.95
Hand1_AllIn_Winnings = 23
Hand1_Fold_Winnings = 3.7
```

Hand 1 Relevant Statistics

Poker is a game of deductive reasoning based on incomplete information. In Hold'em Manager, I can review hands I've played in the past and see the statistics I had at the time the hand was played. Below are screenshots of some of the statistics that I had on this opponent at the time this hand was played.

The numbers represent a specific situation, such as betting preflop from the Button. Most of the time there are two numbers listed. The number on the left indicates the percentage of trials Villain made a specific move in a specific situation. The number in parenthesis on the right indicates the number of trials up to 99 trials. If there is no number in parenthesis, the number of trials is 100 or more.

I had played 392 hands versus this opponent and I've highlighted the stats that are relevant to this hand:

	Total	EP	MP	CO	Btn	SB	BB
VP$IP	27	23	20 (89)	38 (48)	37 (46)	30 (46)	29 (42)
PFR	22	23	19 (89)	21 (48)	26 (46)	26 (46)	14 (42)
Call Open	15	0 (42)	5 (24)	35 (23)	17 (27)	0 (29)	19 (32)
Raise 1st	29	25	22 (60)	36 (22)	53 (15)	62 (13)	
3-Bet	9.2	12.5 (8)	10.0 (20)	4.2 (24)	7.4 (27)	7.7 (26)	13.9 (36)
Limp	0	0	0 (69)	0 (24)	0 (19)	6 (17)	
3bet Vs. Open From		9.6 (73)	2.7 (37)	12.5 (16)	15.4 (13)	50 (2)	
4-bet		7.1 (14)	9 (11)	-	0 (1)	-	0 (2)
Fold to 4-bet		50.0 (2)	Open	4-bet	25.0 (4)		
Fold to Resteal		67 (3)	Squeeze	6.3 (16)	Limp Raise		-

(Loose Aggressive)
Won ($37)
Hands 392
-38 bb/100

1. Villain is in the Button position which is the first position to the right of the small and big blinds. Overall, from this position, Villain 3bets 7.4% (27 trials);

2. Hero is in the Cut-Off position, which is the first position to the right of the Button. Overall, vs. the Cut-Off, Villain 3bets 12.5% (16 trials);

3. When Villain is in the Button vs. a pre-flop raise from the Cut-Off, Villain 3bets 25% (4 trials);

4. When Villain 3bets pre-flop and faces a raise, he folds 50% of the time (2 trials).

Note: How did I decide these 4 points were important? Hero was in the Cut-Off position and Villain 3bet from the Button. I want statistics that show what Villain has done in previous hands in similar situations. How often does Villain 3Bet from the Button position or versus a player in the Cut-Off position? How often does Villain fold a 3Bet to a raise? The stats I selected above were the ones that best answered those questions.

Hand 1 Assumptions

Based on the above statistics, I'm going to make the following assumptions which are educated guesses;

1. Villain raises to 2.50 with about (~) 13-15% of the range of possible starting hands;

2. Villain folds to a re-raise ~ 25% of the time and goes 'All-in' ~75% of the time;

3. Villain re-raises 'All-In' with a ~10% range, which is represented in the image below.

4. The hands highlighted in yellow represent Villain's assumed range, which consists of 128 combinations (9.7% of the sample size).

5. If you're wondering why A5s and A2s are in the range, those represent Villain's bluff hands.

Note: The image below represents the sample size of all possible starting hand combinations in Texas Hold'em, which were 1326 combinations, as you saw in the first part of this tutorial series.

- AA = Ace Ace, KK = King King, etc.

- AKo = Ace King offsuit, ATo = Ace Ten offsuit, etc.

- AKs = Ace King suited, ATs = Ace Ten suited, J8s = Jack Eight suited, etc.

Hand 1 Analysis

Now that you have Villain's range, you need to determine the probability that Hero's hand will win versus Villain's assumed

range if both go all in. You can plug Hero's hand (10h10s) and Villain's range into an equity calculator to determine the probability Hero will win. The equity calculator simulates 10h10s versus Villain's assumed range thousands of times and determines that Hero wins about 53.77% of the time.

Now you can create variables for `EstFoldPercent` and `Equity`, and then create your `AllinExpectedValue()` function.

```
Hand1_Fold_Percent = .25
Hand1_Equity = .538
```

Now it's a simple calculation. You need to build a function that represents the following equation:

$FoldEV = (FoldPercent * FoldWinnings)$

$AllinEV = (1 - FoldPercent) * ((AllInWinnings * Equity) + (AllInLoses * (1 - Equity)))$

$AllinExpectedValue = FoldEV + AllinEV$

```
def AllinExpectedValue(AllIn_Loses, AllIn_Winnings, Fold_Winnings, Fold_Percent, Equity):
    FoldEV = (Fold_Percent * Fold_Winnings)
    AllinEV = (1 - Fold_Percent) * ((AllIn_Winnings * Equity) + (AllIn_Loses * (1 - Equity)))
    AllinExpectedValue = FoldEV + AllinEV
    if AllinExpectedValue > 0:
        return 'Raise!', AllinExpectedValue
    else:
        return 'Fold!', AllinExpectedValue
```

Now plug the Hand 1 variables into the `AllinExpectedValue` function and print your results:

```
Hand1_Decision, Hand1_EV = AllinExpectedValue(Hand1_AllIn_Loses, Hand1_AllIn_Winnings, Hand1_Fold_Winnings, Hand1_Fold_Percent, Hand1_Equity)
print(Hand1_Decision)
print(Hand1_EV)
```

```
Raise!
2.9463250000000016
```

Hand 1 Results

Hero wins the hand with a set (three of a kind) of Tens beating a pair of Kings.

Pocket 8s versus Pre flop Raise

Hand 2 Overview

In this situation, Hero (GamingFoSho) bet \$1.10 and has \$28.58 remaining. Villain, on the other hand, raises to \$3.80 and has \$10.66 remaining. There is currently \$5.40 in the pot. This is a similar situation as the last hand. Again, calling is not a good option. Folding costs nothing, so you need to determine if raising All-in has a positive expected value.

If Hero goes All-in, he Risks \$13.36 since Villain has \$10.66 remaining, and has raised \$2.70. Hero wins \$16.06 if All-in and wins the hand (Villain has \$10.66 remaining, and there is currently \$5.40 in pot) and wins \$5.40 if All-in and Villain folds.

```
Hand2_AllIn_Loses = -13.36
Hand2_AllIn_Winnings = 16.06
Hand2_Fold_Winnings = 5.40
```

Hand 2 Relevant Statistics

1. Villain is in the Small-Blind position. From this position, Villain 3bets 7.2% (100+ trials);

2. Hero is in the Cut-Off position, and vs. the Cut-Off, Villain 3bets 7.9% (89 trails);

3. Villain folds to a 4bet 25% (8 trials);

4. Villain 3bets 4% overall (100+ trails), but vs. Hero only 1% (98 trials). This is a huge red flag, and indicates Villain has a very strong hand;

5. Villain 3bets out of position vs. the Cut-Off 9.8% (51 trials). Out of position means Villain will be first to act in the later betting rounds, and Hero will be second to act.

6. Villain folds to a 4bet when out of position 0% (5 trials). Another huge red flag.

Hand 2 Assumptions

Based on the above statistics, I'm going to make the following assumptions;

1. In comparison to most opponents, Villain 3bets ~8-10% of the time, but appears to 3bet 4 times as often vs. opponents other than Hero. This puts Villain's range at ~3%.

2. Villain has yet to fold to a 4bet in this situation, so at most Villain is folding 10% of the time.

3. Villain calls with a ~2.7% range, which is represented in the image below.

4. The hands in yellow represent Villain's assumed range, which consists of 34 of the 1326 possible combinations of starting hands (2.6%).

Hand 2 Analysis

Once again, I'll use an equity calculator to determine Hero's equity. In this case, Hero's hand has 35.84% equity vs. Villain's range.

Now you can create variables for `EstFoldPercent` and `Equity`.

```
Hand2_Fold_Percent = .10
Hand2_Equity = .3584
```

Now plug the variables for this hand into the `AllinExpectedValue` function, and print the results.

```
Hand2_Decision, Hand2_EV = AllinExpectedValue(Hand2_AllIn_Loses, Hand2_AllIn_Winnings, Hand2_Fold_Winnings, Hand2_Fold_Percent, Hand2_Equity)
print(Hand2_Decision)
print(Hand2_EV)
```

```
Fold!
-1.994284799999999
```

Hand 2 Results

Hero wins the hand with a set of eights vs. two pairs, Aces and Kings. This was a lucky result from a decision that would lose money in the long run. The optimal play here is to fold.

Flop Check Raise Semi Bluff

Hand 3 Overview

This hand is on the flop and Hero (GamingFoSho) is first to act and checks. He has \$25.08. Villain bets \$1.06 and has \$25.37 remaining. There is currently \$2.66 in the pot.

Once again, calling is not a good option, and folding costs nothing. Hero must decide if raising in this situation, known as a Semi-Bluff, has a positive expected value. A Semi-Bluff is when you raise with a drawing hand, in this case a flush draw.

Hero does not have a made hand, as he is currently only holding an Ace high, but has four Hearts and only needs one more for a flush which is a strong hand. Much like the first hand, the best line of play when raising is to raise small and give Villain the opportunity to bluff. If Hero were to just go all in, Villain would most likely fold a hand like King of Hearts/Jack of Hearts.

Hero risks \$25.08 if All-in. Hero wins \$26.68 if All-in and wins hand. Villain bet \$1.06, so Hero must match that amount from the remaining \$25.08, leaving Hero with \$24.02. This means Hero can win an additional \$24.02 on top of the \$2.66 currently in the pot for a total of \$26.68. Hero wins \$2.66 if Villain folds.

```
Hand3_AllIn_Loses = -25.08
Hand3_AllIn_Winnings = 26.68
Hand3_Fold_Winnings = 2.66
```

Hand 3 Relevant Statistics

1. Preflop, Villain raised and from Middle Position, Villain raises 9% (100+ trails);

2. On the flop Hero checks, and Villain (who raised preflop) bets again. This is called a continuation bet (Cbet). Villain Cbets 100% (12 trials);

3. Hero is Check-Raising Villain and when facing a Check-Raise, Villain folds 67% (3 trials);

Hand 3 Assumptions

• Opponents preflop range is ~9-10%, which looks like this:

- The hands highlighted in yellow represent Villain's range, which consists of 128 of the 1326 possible combinations of starting hands (9.7%);

- After Cbeting 100% of the time, Villain folds ~67% of the time to Hero's Check-Raise, leaving ~35-45 combinations in Villain's range. Villain's range looks like this:

- This range consists of 35 combinations and make up the strongest hands from Villain's pre-flop range.

- Some of the hands are different shades of yellow. There are a couple reasons for this;
  1. Card removal - Pocket Aces (AA) normally has six combinations (Diamonds/Clubs, Hearts/Clubs, Spades/Clubs, Hearts/Diamonds, Spades/Diamonds, Spades/Hearts), but Hero has the Ace of Hearts. Villain could only have three combinations of Pocket Aces (Diamonds/Clubs, Spades/Clubs, Spades/Diamonds).

  2. Flush Draws - There are four combinations of King/Jack suited (Diamonds, Hearts, Spades, and Clubs), but Villain would only go All-in with one of those combos (Hearts).

Hand 3 Analysis

Once again, I plug Hero's hand and Villain's range into the equity calculator, this time also entering the 3 flop cards. The equity calculator indicates that Hero wins in this situation 43.58%.

Now you can create your variables for `EstFoldPercent` and `Equity`.

```
Hand3_Fold_Percent = .67
Hand3_Equity = .4358
```

Now plug the Hand 3 variables into the `AllinExpectedValue` function and print the results:

```
Hand3_Decision, Hand3_EV = AllinExpectedValue(Hand3_AllIn_Loses, Hand3_AllIn_Winnings, Hand3_Fold_Winnings, Hand3_Fold_Percent, Hand3_Equity)
print(Hand3_Decision)
print(Hand3_EV)
```

```
Raise!
0.9496126400000008
```

Hand 3 Results

Hero wins with and Ace high flush, beating Villains pair of Tens.

Conclusion

Congrats, you've now completed this tutorial on probability theory with Python! After learning about core probability concepts including Independent/Dependent events, Permutations and Combinations, Multiple events, Expected Values, and how to calculate each of them in the first part of this probability and statistics tutorial series, you applied those concepts to real life poker situations.

In this tutorial, you gained a much deeper understanding of Expected Value and used the concept to make optimal decisions at the poker table. The concept is an extremely useful one to keep in mind when making any business or financial decision, so next time you face such a decision, try to quantify your options in terms of Expected Value.

▲17   💬0