BETTING ON TENNIS

Addison Cassel ~ Mohamad Sahil

University of Pittsburgh
Decision Models
Fall 2016

Contents

Background

People have been placing bets or wagers on races, fights, and lotteries for thousands of years. Throughout this time, the bettors placed their wagers on gut feelings or other non-scientific methods. With the adoption of the internet, sports betting has exploded into a complex web of algorithms and data. Sites like betfair.com and bet365.com provide sports gamblers an avenue for placing betting on all kinds of sports all over the world. Naturally this was an interesting model for us to study using the mathematical techniques learned in Decision Models.

The area we chose to study specifically was tennis betting. Tennis was a great subject for this project because bets in tennis are much simpler than other sports. Specifically, there are only two outcomes. Either one player wins or the other one does. There aren't any point spreads or other tools for an odds maker to use. Only the two participants and the odds that each player might win.

To place this model in the real world we decided to model the final three rounds of the 2016 Wimbledon Gentlemen's Singles tournament. This was an interesting subject in particular because this year's quarterfinals had a lot of uncertainty. World number one at the time, Novak Djokovic was surprisingly defeated by Sam Querrery in the third round who was unexpectedly entered into the quarterfinals. Other long shots were present as well including Lucas Pouille. All of these factors created a quarterfinal bracket that was highly unexpected and highly uncertain. This is a great environment to test our betting system.

Goal

Our goal is to generate a realistic simulator to test multiple heuristic policies and determine a lucrative betting policy based on known probabilities and simulated odds. We want to maximize our winnings over the course of these three rounds.

Assumptions

In order for this model to work we will have to make a few very large assumptions. The most important assumption we will have to make is that the true probabilities of a player winning a match against another player is known to the bettor. Without this assumption we will not be able to effectively place bets. These probabilities were created by our team with the idea that while the probabilities may not actually be exactly accurate they will be close enough to create a realistic and useful model.

Winning Probabilities for Each Possible Matchup

		Loser							
	Player	Querrey	Federer	Berdych	Tsonga	Raonic	Cilic	Puille	Murray
Winner	Querrey	1	0.2	0.29	0.32	0.21	0.4	0.55	0.14
	Federer	0.8	1	0.62	0.65	0.56	0.59	0.75	0.42
	Berdych	0.71	0.38	1	0.51	0.44	0.48	0.65	0.11
	Tsonga	0.68	0.35	0.49	1	0.42	0.43	0.64	0.22
	Raonic	0.79	0.44	0.56	0.58	1	0.55	0.72	0.36
	Cilic	0.6	0.41	0.52	0.57	0.45	1	0.71	0.31
	Puille	0.45	0.25	0.35	0.36	0.28	0.29	1	0.02
	Murray	0.86	0.58	0.89	0.78	0.64	0.69	0.98	1

Another assumption that we will have to make is that the odds we are betting on will have some random error associated with them but the odds will be known at the time of betting. Later on when the odds are known our heuristic approach will bet on matches with a large error from the true probability.

Model Formulation

We proceeded to build a simulator that will create odds for each round's matches, simulate winners for each match in a round, and fill out the next round's bracket. First we assigned each player a number. This was important to be able to correctly index each match accurately. There are only certain number of possible semifinal matches (the winner of Querrey vs. Raonic playes the winner of Federer vs. Cilic for example) and these index values will be stored in each rounds arrays in order to access each player's winning probability. The following list shows each players index value.

Player Index Values

1. Querrey
2. Federer
3. Berdych
4. Tsonga
5. Raonic
6. Cilic
7. Puille
8. Murray

We decided to use arrays of sizes 4x2, 2x2, 1x2 to store the matchups for the quarterfinals, semifinals, and finals respectively. The simulator will access the probability matrix above to generate probabilities with some normal random error and then simulate match outcomes using a uniform random variable. An example of how this update works is shown below for the quarterfinal matches.

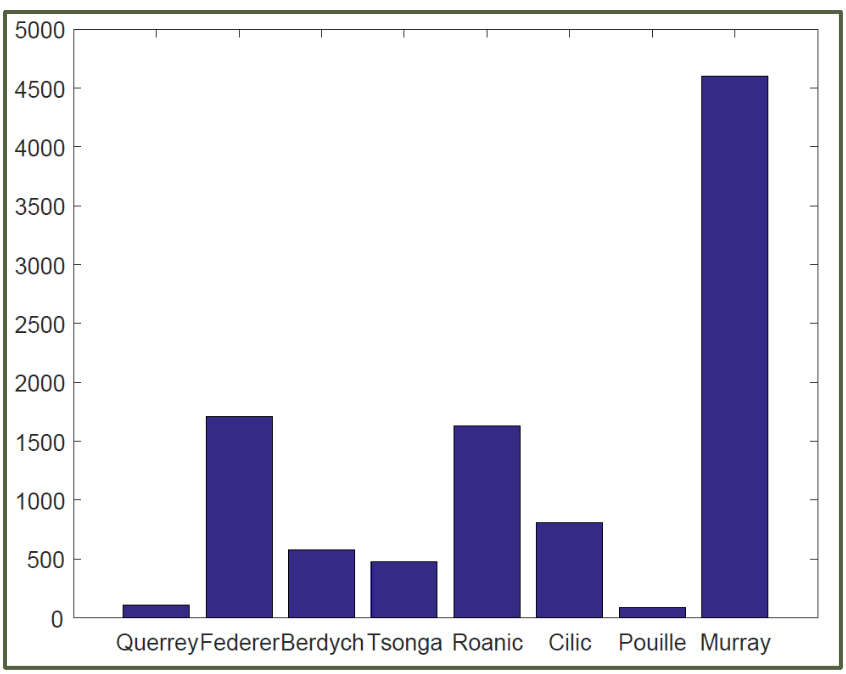
```
    %%%% Simulate Quarterfinal Matches
    rand_q=rand(4,1);

    for m=1:4

        if rand_q(m)<=prob(matches_q(m,1),matches_q(m,2));
            matches_s(m)=matches_q(m,1);
        else
            matches_s(m)=matches_q(m,2);
        end
    end

matches_s=transpose(matches_s);
```

We validated our model by identifying who would have won the tournament for 10,000 simulations. We found that Andy Murray won nearly half of the simulated tournaments which makes sense given that he was the 2016 Wimbledon Champion and the world number two ranked player at the time. Federer and Roanic also had a comparatively large number of wins, and each won about the same amount of times. A bar graph of the all the results is shown below.



Summary of Model Formulation

State Variables

- Current amount of money (M_t) $M_1 = 1{,}000$
- Current matchups (G_t) $G_1 = \{1,5;2,6;3,7;4,8\}$
- Current match odds (O_t)

Decision Variable

- Bets(B_{ijt})

Constraints

- Sum of bets for each round must be less than the current balance
- $\sum_1^i \sum_1^j (Bij)t \leq Mt$ (For all rounds t)

Exogenous Information

- Match outcomes (G_{t+1})

Update Equations

- $Mt + 1 = Mt - \sum_1^i \sum_1^2 Bij + \sum_1^i \sum_i^j [B(ij)(Oij) * INDICATOR((Gij(t) = Gij(t+1))]$
  - Each rounds balance is the previous balance plus the winnings of the previous round minus the current bets. We multiply this by an indicator function that is 1 if that player advanced and 0 if they did not.
- $O_{ij}=(1/P_{if}) * \varepsilon - R$
  - Odds are generated for each round but they are known at the time of betting and not exogenous to our decision.

Objective

- Max M_T
  - Maximize the final balance at the end of the tournament

Heuristic Policies

Approach

We decided to test multiple heuristics to maximize our expected profit. In this case we will calculate the expected profits by taking the mean of the results of 10,000 simulations. This will be our primary indicator for the model's success. There will be other factors that an individual will want to take into account when placing bets such as the model's variance, frequency of busts and frequency of wins but to compare two models against each other we will focus primarily on the mean ending balance.

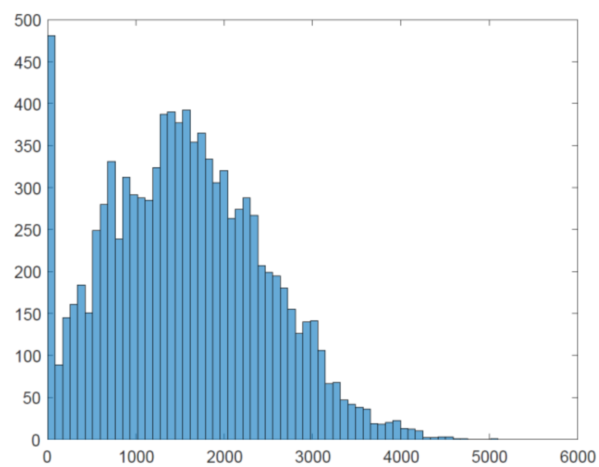
First Policy

For our first heurist policy we decided to place a bet of either nothing, a small amount, or a large amount on a player to win a match. This policy places small bets where the expected value of our bet is larger than zero and large bets when the bet is significantly larger than zero. These values were calculated by taking the difference of the reciprocal of the odds and the winning probabilities. When this value is greater than zero we will place a small bet, and when this amount is greater than 0.4 we will place a large bet. Multiple bet sizes were tested in order to tune this policy to the highest expect value.

Summary of Outcomes Policy 1

Trial	Small Bet	Large Bet	Mean
1	100	500	1427
2	200	500	1419
3	300	500	1388
4	50	500	1402
5	75	500	1451
6	75	525	1611
7	75	550	1609
8	75	575	1632
9	75	600	1692
10	75	625	1698
11	75	650	1706
12	75	675	1741
13	75	700	1697
14	75	725	1641

The outcomes of this policy show good promise and certain bet amounts produced high expected returns. After testing multiple values for our low and high bet it became clear that comparatively small low bets and large high bets produced better returns. One potential issue with this policy is that it does not take into account the amount of times the bettor would completely bust or lose all of the starting balance over the course of betting. A histogram of our results is shown below and reflects this issue. Some individuals may prefer a more conservative approach even though this policy has a large expected return.



Second Policy

Our second policy is similar in structure to our first policy. There are low bets and high bets based on the same threshold for when to place them. The difference is in this policy we will increase the amount of bet that we are placing in each round. The small bet will increase by $50 each round and the large bet will increase by $100 each round. The motivation behind this policy is that it allows the bettor to take advantage of his or her expected winnings in the previous rounds of betting. Again a range of values for the bet amounts was tested below. The bet sizes here represent the amount bet in the quarterfinals and they increase for the next two rounds. We started with a conservative formulation of the first heuristic to begin this trial.

Trial	Small Bet	Large Bet	mean
1	75	500	1689
2	100	550	1688
3	125	600	1718
4	150	650	1692

The outcome of these trials show that this policy is not as promising as we thought before running the simulations. Overall we could not get this policy to perform as well as the simple static bet policy. This may be because the overall balance did not increase enough each round to justify us raising the betting amounts at the rate we did.

Third Policy

After testing the first two heuristics it became clear that we would want a policy that produced bets of sizes other than the predetermined amounts above. We determined that it would make sense to produce bets each round based on a fraction of our total balance at that time. We decided to keep the two threshold structure of our model in place but just change the small and large bets to proportions of the round's starting balance. The motivation for this was that a big win in an early round produces much more earning potential so as the model runs we would expect a larger maximum value each round and also a larger variance. This is still attractive given that we are primarily focused on maximizing expected profits. The table below represents some of the values we tried for each of these proportions.

Trial	Low %	High %	Mean	Std Dev
1	10%	50%	1673	
2	15%	50%	1664	1384
3	7%	50%	1704	1391
4	7%	25%	1305	599
5	5%	50%	1742	1396
6	5%	25%	1309	551

The outcome of this trial produced good results with a larger ceiling. This policy makes the most sense because bet amounts are determined by the starting balance and not by some arbitrary tunable amount like in the first two policies.

Final Policy

The final policy we decided to go with is based on our third heuristic. We produced two policies that a better may find advantageous depending on their own goals. The first is our aggressive policy this policy increases expected profits by placing larger bets but also incorporates more risk of loss. The second policy reduces expected profits but with the tradeoff that there is less variation and the bettor comes out ahead more frequently. A tabulation of these policies and their outcomes is shown below (win% is the amount of times a bettor comes out with more money than he or she started with).

	Low Bet	High Bet	Mean	Std Dev	Win %
Aggressive	5%	50%	1742	1396	64%
Conservative	5%	25%	1309	551	68%

Upper Bound

The theoretical optimum for this project would simply be placing a bet equal to your current balance on the person who won against the longest odds in a given round. This optimum would require you to know the future so to speak. Obviously no heuristic policy will be able to achieve similar results. Our optimum policy simulation produced a theoretical optimum of $38,039 while our aggressive policy only produced a final balance of $1,742. The rate of return for our aggressive policy was 4.4% of this theoretical maximum.

Conclusions

In general, this model works given our stated assumptions. However, the assumption that we would be able to know the true probability of a player winning would require a lot of confidence. Developing these probabilities would require a lot of simulation with real world data. That was outside of the scope of our project. What our project shows is that if you know a distribution of true probabilities you will be able to produce a positive expected value. Ultimately the exact policy selected is dependent upon your goals as a bettor. We were able to identify two such policies that produce good results depending on how aggressive you want to be.

In a more general sense, this project shows that simulating a situation with known distributions can be incredibly helpful for solving real world problems. Our scope was relatively small as we constrained the model to three rounds of a tennis tournament but the conclusion is the same. Reaching decisions mathematically should always be a priority and if you only use guesses and gut feelings the house will always win.