CS224M

Scoring Methods

Ido Milstein
ido@cs.stanford.edu

1 Introduction

A common setting in the approach that game theory takes, in 2 player games, is to assume that each player has a strategy, and to then find out which strategy does better, if both strategies play each other. The strategies are further evaluated by checking how well they do when playing in some population of other (or same) strategies. Sometimes, the strategies are even compared in dynamic systems of reproduction and/or evolution.

In this paper, instead of evaluating the dynamics of the strategies, I simply wish to address the following question: given a set of *players* , and a set of games played between pairs of them, such that each game is a zero sum game with scores only for victory (1) or defeat (-1), how can we rank the players according to performance? Notice that I used the word *players* rather than strategies. This is because I do not want to assume that every player uses a single strategy, or even a mixed strategy. So, I make no assumptions about convergence of the distribution of strategies that a player uses as the number of games grows.

Some questions arise quite immediately. First, what is performance? Let's say that players A, B and C play a set of games, where each pair plays 5 games. Now assume that A won over B all 5 games they played, C won over B 3 of the 5 games they played, and C won over A 3 of the 5 games they played. Now who is the best player? Clearly not B. C won both duels that he played against B and A (both at a 3:2 ratio), so this may imply that C is the best player. But, A won more games than C (A won 5+2=7 games while C won 3+3=6 games).

Second, assuming that we have defined performance, can we always measure it? What if we have hundreds or thousands of players, and each one of them chooses to play only a small number of games, and only plays an even smaller number of other players. How can we evaluated which player is better if two players never played each other?

2 A closer look at both questions

2.1 Performance

Let us look again at the example above, and try to get a better feeling for what may be a good definition for performance. Assume those three players A, B and C, played many games, in which A won over B 100% of the games, C won over B x%, and C won over A y% of the games. Now assume both $x > 50\%$ and

$y > 50\%$ but,

$$\text{A's victories} = 100 + (100 - y) > x + y = \text{C's victories}$$

. The problem as it was stated in the introduction still remains - on the one hand, C won his matches against both A and B and so should be declared the best player, but on the other hand, A won a greater percentage of his games, and so should be declared the best instead of C.

Assume now that x and y are both just a little above 50%. This means that C is about as good as B in their duels, and C is about as good as A in their duels, and it's only a very small fraction of the games that made him win those matches by over 50%. But, looking at A's scores, we see that A won all his games against B, and was just about as good as C. This gives a strong incentive to the next definition:

Definition:

Let P_A be a Uniform distribution over all players except A, and assume player A plays 1 game against a player that is randomly chosen according to P_A

$$performance_A \stackrel{def}{=} E_{P_A}[P(\textit{player A will win against the chosen player})]$$

So, in our A, B, C example,

$$performance_A = 0.5 * 1 + 0.5 * (1 - \frac{y}{100}) \simeq 0.75$$

$$performance_C = 0.5 * \frac{x}{100} + 0.5 * \frac{y}{100} \simeq 0.5$$

A representative scoring method will be defined as any monotone function of performance.

2.2 Learning and estimating performance

We can now look at the problem in the following way. We have a directed complete graph, the nodes in the graph represent players, and a directed edge from node A to node B represents the probability that player A wins if he plays against player B. We can remove from the graph all edges with values ¡ 0.5, because they add no information. So the only edges we look at point from the probable winner to the probable looser and not vice versa. If we have this graph then a performance of a player is just the average value of the edges that connect to it, where for edges that go into it we take (1-value) instead of (value). Our problem is therefore to find the values and directions of the edges in this graph, given only limited information about sets of games.

At this point it is important to remember that if we want to make inferences that go beyond the given data, we have to make some assumptions. These assumptions are the inductive bias that we bring with us. Bringing a strong bias limits our model but it also lets us make more inferences. If the bias is close to the correct model we get good approximations to the correct model, but this may also exclude the exact model from our scope.

In the following sections I examine several such biases that may be appropriate to the real world problem, starting from no biases at all, and adding more of them in time. So the models I look at keep getting smaller. I will also try to motivate the assumptions I make as I go along.

3 Possible assumptions

3.1 No assumptions

How can we compare 2 players that have never played each other? A simple answer could be to just see what the winning rate of each one of them is, and say that the one that has a higher ratio is better. This is not a good solution, because a good player that chooses to play only against good players, will have a lower winning rate than an intermediate player that only plays weak players. Going back to the inductive bias discussion above, it should be clear that with no assumptions, we can not make any inferences as to which player is better if they never play each other. What about players that did play each other? Maximum likelihood in this case will give us the obvious expected results.

Let us define θ_{ij} to be the probability that player i wins when playing against player j. W_{ij} will be defined as the number of victories i had over j, and L_{ij} as the number of losses. Let G denote our graph, θ the whole set off probabilities, D the data, d[m] the result of game m, and n the number of players. Assume that the game results are independent we get:

$$\begin{aligned} P(G,\theta|D) &\propto P(D|G,\theta) * P(\theta|G) * P(G) \\ &\propto P(D|G,\theta) = \prod_{m=1}^{M} P(d[m]|G,\theta) \\ &= \prod_{i>j} \theta_{ij}^{W_{ij}} * (1-\theta_{ij})^{L_{ij}} \end{aligned}$$

Notice that I assumed uniform priors for graphs and θs so as not to incorporate any prior knowledge about the players. This is maximized when for every pair i,j of players that played each other at least once $\theta_{ij} := \frac{W_{ij}}{W_{ij}+L_{ij}}$. As we can see this doesn't depend on θ_{ij} if i and j never played each other.

3.2 Transitivity

One assumption that seems quite plausible is that if player i wins when playing player j, and j wins when playing player k, then i should win when playing against player k. This assumption is not always a good one, and one can find many cases in which it fails. But, it is usually a good approximation to the correct state. This assumption can be formulated like this:

$$[\theta_{ij} > 0.5 \wedge \theta_{jk} > 0.5] \Longrightarrow \theta_{ik} > 0.5$$

Notice that this does **not** mean that player i has the highest performance, as Figure 1 shows. In the figure, even though player i wins over both j and k, and even though we have transitivity, player j is the one that has the highest performance ($performance_i = (0.6{+}0.6)/2$ and $performance_j = (1.0{+}0.4)/2 = 0.7$).

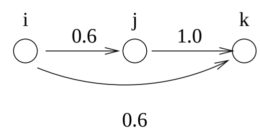

Figure 1: Winning probabilities for three players

If none of the edges have an exact 0.5 value - this makes our graph G an acyclic graph and so we can define a topological order on it. If 0.5 edges do exist we can always reverse their order to comply with a topological order of the other edges. More over,

Proposition:

If an undirected graph G is a clique, then for every topological order on the nodes of G there is a unique redirection of the edges of G.

Proof:

If T1 and T2 are two different topological orders on G, there there exist nodes A, B such that A appears before B in T1 and B appears before A in T2. G is a clique, and therefore includes an edge $A \leftrightarrow B$. So, redirecting it according to T1 will result as $A \rightarrow B$ and redirecting it according to T2 will result as $A \leftarrow B$ ■

Note that every directed acyclic clique has a directed Hamiltonian path in it, and so for every such graph there is a **unique** topological order. From this and from the above proposition, we get that our acyclic cliques are in a 1-1 correspondence with topological orders. So, there are n! equally possible graphs under our assumption, instead of the 2^n possible graphs that exist without the assumption. Notice that

$$P(\theta_{ij} \geq 0.5|G) = \begin{cases} 1 & (i \rightarrow j) \in G \\ 0 & (j \rightarrow i) \in G \end{cases}$$

and so we get the conditional independence

$$\forall_{ijkl} \qquad P(\theta_{ij}, \theta_{kl}|G) = P(\theta_{ij}|G) * P(\theta_{kl}|G)$$

with this independence in mind, we get that

$$\begin{aligned} P(G,\theta|D) &\propto P(D|G,\theta) * P(\theta|G) * P(G) \propto P(D|G,\theta) * \prod_{\substack{i>j \\ topo'}} P(\theta_{ij}|G) \\ &= \prod_{m=1}^{M} P(d[m]|G,\theta) \prod_{\substack{i>j \\ topo'}} P(\theta_{ij}|G) \\ &\propto \prod_{i>j} \theta_{ij}^{W_{ij}} * (1-\theta_{ij})^{L_{ij}} * I_{[(\theta ij \geq 0.5 \wedge (i \to j) \in G) \vee (\theta ij \leq 0.5 \wedge (j \to i) \in G)]} \end{aligned}$$

The meaning of this is that given the graph, each of the θs is maximized separately. Now, for every pair of nodes i, j for which there is an edge $i \to j$, we know that $0.5 \leq \theta_{ij} \leq 1$. So if $W_{ij} \geq L_{ij}$ there's no problem and again we have that the best estimate is $\widetilde{\theta_{ij}} := \frac{W_{ij}}{W_{ij}+L_{ij}}$. But, if $W_{ij} \leq L_{ij}$ then the best estimate $\widetilde{\theta_{ij}}$ that is still in the [0.5, 1] range is just 0.5. The "value" that every edge brings towards to total likelihood is therefore (I used log to make it additive)

$$v(i,j) = W_{ij} log(\widetilde{\theta_{ij}}) + L_{ij} log(1- \widetilde{\theta_{ij}})$$

The problem we face now is: Assume that we have an undirected clique G, with values for edges $v(i,j)$ and $v(j,i)$ (corresponding to the possible redirections of the edge $i \leftrightarrow j$). From all possible redirections of all edges of G, such that the redirected graph is acyclic, pick the one that has the highest score, where a score of a redirection is

$$\sum_{(i \to j) \in G} v(i,j)$$

I couldn't find a polynomial time algorithm to solve this, nor have I been able to prove that it is NP-hard (maybe if I had a little more time...), so I guess that my best suggestion for this case would be to use the common local search methods such as "hill climbing" and others.

It is important to note, that if we add a new player (node), with an unbounded number of games played against the current players, the entire order may change into **any** other order (easy to prove). But, if increments are allowed only one game at a time, then changes in the order would be small (especially after many games have already been played), and so it should be possible to keep track of the order by switching the order of adjacent players one pair at a time.

In any case $P(\theta_{ij} > 0.5|D)$ for players i, j that never played each other could be approximated by the chose graph as 0 or 1 (depending if i appears after j in the topological order, or vice versa), or calculated explicitly by

$$P(\theta_{ij} > 0.5|D) = \sum_{G} P(G|D) * I_{[(i \to j) \in G]}$$

3.3 Transitivity and monotonicity

A slightly stronger assumption than the previous one, is that if player i usually wins when playing against j, then when playing against other players he'll win more than j will. This can be formulated as:

$$\theta_{ij} \geq 0.5 \Longrightarrow \forall_k \;\; \theta_{ik} \geq \theta_{jk}$$

Proposition:

Under the monotonicity assumption, the first player on the topological order has the best performance.

Proof:

Note that we can talk about ***the*** *order because we proved before that it is unique under a weaker assumption. Let 1...n be the topological order on G, and let k be some player in that order other than 1.*

$$
\begin{aligned}
performance_1 &= \frac{1}{n-1}\sum_{j\neq 1}\theta_{1j} = \frac{1}{n-1}[\theta_{1k} + \sum_{j\neq 1 \wedge j\neq k}\theta_{1j}] \\
&\text{From monotonicity and knowing } \theta_{1k} \geq 0.5 \text{we get} \\
&\geq \frac{1}{n-1}[\theta_{1k} + \sum_{j\neq 1 \wedge j\neq k}\theta_{kj}] \\
&\geq \frac{1}{n-1}[\theta_{k1} + \sum_{j\neq 1 \wedge j\neq k}\theta_{kj}] \\
&= \frac{1}{n-1}\sum_{j\neq k}\theta_{kj} = performance_k \qquad \blacksquare
\end{aligned}
$$

Let's look at the maximum likelihood results for the model under the monotonicity assumption, this time we'll do it in 2 steps. First note that if the topological order of G is 1..n, and θ satisfies the monotonicity assumption then we get the following density function:

$$
\begin{aligned}
P(\theta|G) &= P(\theta_{12}|G,\theta_{13},\ldots,\theta_{1n},\theta_{23},\ldots,\theta_{(n-1)n}) \\
&\quad * P(\theta_{13},\ldots,\theta_{1n},\theta_{23},\ldots,\theta_{(n-1)n}|G) \\
&= P(\theta_{12}|G) * P(\theta_{13},\ldots,\theta_{1n},\theta_{23},\ldots,\theta_{(n-1)n}|G) \\
&= P(\theta_{12}|G) * P(\theta_{13}|G,\theta_{14},\ldots,\theta_{1n},\theta_{23},\ldots,\theta_{(n-1)n}) \\
&\quad * P(\theta_{14},\ldots,\theta_{1n},\theta_{23},\ldots,\theta_{(n-1)n}|G) \\
&= P(\theta_{12}|G) * P(\theta_{13}|G,\theta_{23}) \\
&\quad * P(\theta_{14},\ldots,\theta_{1n},\theta_{23},\ldots,\theta_{(n-1)n}|G) = \ldots \\
&= [P(\theta_{12}|G) * \prod_{j>2} P(\theta_{1j}|G,\theta_{2j})] * P(\theta_{23},\ldots,\theta_{2n},\theta_{34},\ldots,\theta_{(n-1)n}|G)
\end{aligned}
$$

$$
\begin{aligned}
&= [P(\theta_{12}|G) * \prod_{j>2} P(\theta_{1j}|G,\theta_{2j})] * [P(\theta_{23}|G) * \prod_{j>3} P(\theta_{2j}|G,\theta_{3j})] * \ldots \\
&\quad * P(\theta_{34},\ldots,\theta_{3n},\theta_{45},\ldots,\theta_{(n-1)n}|G) = \ldots \\
&= \prod_{i=1}^{i=n-2} \prod_{j>i+1} P(\theta_{ij}|G,\theta_{(i+1)j}) * \prod_{i=1}^{i=n-1} P(\theta_{i(i+1)}|G) \\
&= \prod_{i=1}^{i=n-2} \prod_{j>i+1} \frac{1}{1-\theta_{(i+1)j}} * \prod_{i=1}^{i=n-1} P(\theta_{i(i+1)}|G) \\
&= \prod_{1<i<j\leq n} \frac{1}{1-\theta_{ij}} * \prod_{i=1}^{i=n-1} P(\theta_{i(i+1)}|G) \\
&= \prod_{1<i<j\leq n} \frac{1}{1-\theta_{ij}} * 2^{n-1}
\end{aligned}
$$

If we go on now to look at the likelihood of a model we get (assuming again that θ satisfies monotonicity on G):

$$
\begin{aligned}
P(G,\theta|D) &\propto P(D|G,\theta) * P(\theta|G) \\
&\propto \prod_{1\leq i<j\leq n} \theta_{ij}^{W_{ij}}(1-\theta_{ij})^{L_{ij}} * \prod_{1<i<j\leq n} \frac{1}{1-\theta_{ij}} * 2^{n-1} \\
&\propto \prod_{1<i<j\leq n} \theta_{ij}^{W_{ij}}(1-\theta_{ij})^{L_{ij}-1} * \prod_{1<j\leq n} \theta_{1j}^{W_{1j}}(1-\theta_{1j})^{L_{1j}}
\end{aligned}
$$

So, this is practically like finding the best fitting parameters for each graph, but for every pair $i \to j$ in G such that $i \neq 1$ we disregard 1 game in which i lost. This shows us that in this case too the "score" for each graph can be computed easily, but as before the problem of how to search over the space of graphs remains. (I hope I got the math right...)

3.4 Strong parametric dependencies

All the cases I talked about so far regarded the problem of finding a scoring method as a static problem. I would like to address now the dynamics of a scoring method as player go on playing games.

More specifically, one may like to have a scoring method that gives absolute scores, that don't vary with time if a player stops playing. Note that this is not the case in general for performance measures as I defined them above, because the population of players may change, and with it the expected performance of a player even if he stops playing. This means that if we add the result of a game to a current list of results, this should not change the scores of any of the players that didn't play in that game.

Notice that this assumption means that the scores of 2 players uniquely define the probability distribution of victories in games between them. Let $f(s_A, s_B) := \theta_{AB}$ denote the function that takes the scores of 2 players to the probability that the first wins a game between them. I will use the shorthand

θ_{xy} to mean the probability that a player with score x will win a player with score y, this is well defined because of the above uniqueness conclusion.

We can start by making one very convenient assumption, that new players are chosen from a stationary distribution. This means that if a player wishes to play a "newbie", than the expected number of victories he will have in these games will not vary with time. We can now go on and arbitrarily define the score for the average "newbie" to be 0. Let a randomly chosen "newbie" be denoted by 'N'. We get that for player i with score x

$$f(x,0) = \theta_{iN}$$

Note that knowing θ_{iN} does not yet limit the score of player i to a single value, because $x \neq y \not\longrightarrow f(x,0) \neq f(y,0)$.

At this point I would like to make an additional plausible assumption that will simplify the problem of finding and learning such an f even further. The assumption I would like to make, is that

$$\theta_{ik} = \frac{\theta_{ij}\theta_{jk}}{\theta_{ij}\theta_{jk} + (1-\theta_{ij})(1-\theta_{jk})}$$

Here are 2 similar justifications for this assumption. First, this is like saying that instead of playing each other, player i and k have to play with a mediator j. If i wins j and j wins k then we say that i won over k, and vice versa. cases in which j won both games or lost both games are ties between i and k and are there fore disregarded. Second, if we look at the proportions of victories for the three player, we have $i : j = W_{ij} : L_{ij}$ and $j : k = W_{jk} : L_{jk}$. if we bring these to a mutual basis we get $i : j : k = W_{ij}W_{jk} : L_{ij}W_{jk} : L_{ij}L_{jk}$ and therefore

$$\begin{aligned} \theta_{ik} &= \frac{W_{ij}W_{jk}}{W_{ij}W_{jk} + L_{ij}L_{jk}} = \frac{W_{ij}W_{jk}}{W_{ij}W_{jk} + L_{ij}L_{jk}} * \frac{(W_{ij}+L_{ij})(W_{jk}+L_{jk})}{(W_{ij}+L_{ij})(W_{jk}+L_{jk})} \\ &= \frac{\theta_{ij}\theta_{jk}}{\theta_{ij}\theta_{jk} + (1-\theta_{ij})(1-\theta_{jk})} \end{aligned}$$

If we join this new assumption with our previous definition for $f(x,0)$, we get

$$f(x,y) = \theta_{xy} = \frac{\theta_{x0}\theta_{0y}}{\theta_{x0}\theta_{0y} + (1-\theta_{x0})(1-\theta_{0y})} = \frac{\theta_{x0}(1-\theta_{y0})}{\theta_{x0}(1-\theta_{y0}) + (1-\theta_{x0})\theta_{y0}}$$

The conclusion from this is that under this assumption f is completely defined by the values that it gives $f(x,0)$ for all x's. Now all that is left for us to do, is to pick some monotone function for $f(x,0)$, and to find a good learning rule that will converge to it. Picking such a function is a matter of convenience. Learning it is an issue that even though I already have several ideas about, I haven't finished formulating these ideas, and will therefore not include in this paper.

Some issues that arise about this are - should the scores have a zero sum? how can we adjust the scores so that they take into account the fact that players get better (and sometimes worse) with time? Should games that were played a long time ago be considered as important as recent games?

4 Conclusion

Since I couldn't find any previous research on this problem, my results are in some cases elementary, and in some cases more interesting. I have tried to look at different possible models and assumptions that we can make about the relations between player's strengths, and see what we can learn from them. On the way I came across one problem for which I couldn't find a polynomial time algorithm, or and NP-hardness proof. I have proved for several cases that calculating the likelihood of a model can be separated into likelihoods of single edges in a graph, and found some nice closed form solutions for them. My guess is that if one wanted to go on and further investigate the problem of creating scoring systems, the important direction would be the last one I only briefly addressed, which is how can we learn dynamic scoring system efficiently? and what strong assumptions can we use to do that?

References

None