OPERATIONS RESEARCH
Vol. 55, No. 6, November–December 2007, pp. 1163–1177
ISSN 0030-364X | EISSN 1526-5463 | 07 | 5506 | 1163

DOI 10.1287/opre.1070.0448

Optimal Strategies for Sports Betting Pools

Bryan Clair, David Letscher
Department of Mathematics and Computer Science, Saint Louis University,
St. Louis, Missouri 63103 {bryan@slu.edu, letscher@slu.edu}

Every fall, millions of Americans enter betting pools to pick winners of the weekly NFL football games. In the spring, NCAA tournament basketball pools are even more popular. In both cases, teams that are popularly perceived as "favorites" gain a disproportionate share of entries. In large pools there can be a significant advantage to picking upsets that differentiate your picks from the crowd.

In this paper, we present a model of betting pools that incorporates pool participant behavior. We use the model to derive strategies that maximize the expected return on a bet in both football pools and tournament-style pools. These strategies significantly outperform strategies based on maximizing score or number of correct picks—often by orders of magnitude.

Subject classifications: games/group decisions: gambling.
Area of review: Optimization.
History: Received March 2006; revisions received July 2006, September 2006; accepted October 2006.

1. Introduction

In a betting pool, players pay a fixed bet to make predictions about future events, and the pooled bets are paid to the player or players whose predictions prove to be most accurate. In sports betting pools, there is often a disconnect between the fraction of contest entrants choosing a team, and that team's actual probability of winning. Frequently, this takes the form of an "overperception of the favorites," where a team with a slight edge is picked by a large majority of pool entrants.

In March 2003, for example, approximately one million people entered ESPN's Tournament Challenge, an online contest to predict the outcome of the NCAA Men's Basketball Tournament. That year, Kentucky finished the regular season with a 23-game winning streak and was the clear favorite, but the NCAA tournament is a 64-team single-elimination tournament notorious for upsets. Nevertheless, 51% of the ESPN pool participants predicted the Kentucky Wildcats as champion.

Kentucky lovers faced an uphill battle to win that pool; they needed Kentucky to win and then still had to beat about a half million other entrants at picking the rest of the games. Potentially, a better strategy for winning was to pick an underdog champion and hope to be part of a much smaller group—only 25,000 entrants correctly chose eventual champion Syracuse.

Less important but equally striking were the four "8-9" games that year, which match the two middle seeds in each group of 16. Historically, these games are toss-ups, but in all four matchups the Tournament Challenge entrants had anointed one team the favorite by at least a two-to-one margin.

This phenomenon has not gone unnoticed. In a limited study of pools for the 1993 NCAA tournament, Metrick (1996) concluded that #1 seeds (the top four teams) were overbacked by pool entrants, and that possible profit opportunities were available for betting lower seeds.

Betting on weaker teams (underdogs) is generally not the way to achieve a high average score. However, betting pools are about winning a share of the pot. That is, there is a crucial distinction between maximizing expected score and maximizing expected return. A good score is worthless if most of the pool entrants also score well, whereas a mediocre score can win a pool when many games are upsets. In a large pool, picking extra underdogs can substantially increase the chances of a first-place finish and a return on the bet. The subtle problem is to find the balance between choosing high-probability events (betting on favorites) and going against the crowd (betting on underdogs).

The sports betting pool is an instance of a more general competitive setting, in which competitors must select from among a number of choices subject to two broad conditions: first, that all competitors are in general agreement as to which choices are desirable; second, that the benefits of a particular choice decrease with an increase in the number of competitors making that choice.

As an example, airline routes with high passenger traffic are desirable for their high revenue. However, as the number of airlines flying a route increases, the competition for airport gates and ticket-pricing battles decrease the benefits to each airline.

Another example is the combinatorial auction, in which every bidder bids on packages of items requiring an all-or-nothing outcome. Bids for packages that include the most desirable items will face more competition and are less likely to be satisfied.

Sports betting pools provide a model example of such a situation, and have the advantage that the key variables are readily measurable, allowing for precise quantitative analysis. One is able to test and verify the theoretical model, giving concrete evidence that less desirable choices can in fact lead to practical advantages in a competitive situation.

Previous works (Kaplan and Garstka 2001, Breiter and Carlin 1997) discuss finding NCAA tournament picks that maximize one's expected score, but are not concerned with opponent behavior and therefore miss the competitive nature of the situation. More recently, Kaplan and Magazine (2003) present a simple model of opponent behavior, but in the context of a nontraditional NCAA basketball pool where entrants bid to "own" certain teams.

In this paper, we propose a complete probability model for betting pools (§2) that applies to traditional NFL football pools and NCAA tournament-style pools. The model incorporates probable game outcomes as well as information about pool participant behavior. We consider the optimization problem of finding picks $\mathbf{b}$ that maximize expected return $E(\mathbf{b})$ on a bet of a fixed amount. One of the more appealing aspects of the problem is that solutions are sensitive to the number of opponents in the pool, generally picking more conservatively in small pools and choosing more upsets in larger pools.

For football pools, we give an an exact formula for the expected return $E(\mathbf{y})$ for any picks $\mathbf{y}$ (Theorem 3.3). Exhaustive search or other standard search algorithms can then find optimal picks $\mathbf{b}$.

We applied these techniques to a number of online NFL pools for the 2004–2005 season (§3.3). We found that pool participants overbet favorites, leaving plenty of room for strategic improvements. In larger (thousands of players) pools, crowd avoidance is essential—picking all favorites is one of the few losing bets.

For tournament pools, such as NCAA basketball, Theorem 4.1 and §5.2 describe a method to approximate expected return. The approximation relies on the observation that pool participant scores are approximately normally distributed. One can then search for picks that maximize the approximate value.

We were able to use these techniques to analyze (retroactively) the 2004 NCAA men's basketball tournament and to enter NCAA pools in 2005. The best picks we found had expected returns that were orders of magnitude better than score-maximizing strategies. The results, in §5.4, mirror the results with football pools in that larger pools call for more upset picks. It also appears (e.g., Table 5) that one should pick conservatively in early rounds, choosing upsets in later rounds and the final four.

2. The Pool Model

In a sports betting pool, participants attempt to predict the winners of a collection of sporting events, such as football or basketball games. Their predictions, known as "picks," are given to the pool organizer, along with a fixed bet. After the games are played, each participant receives a score, with points awarded for correct predictions. Players with the most points receive prizes or a share of the pooled bets.

In this section, we describe a simple probability model for a sports betting pool that encompasses participant behavior, game outcomes, and pool payoff schemes. We then state the optimization problem addressed by this paper.

2.1. Pool Payoff Schemes

Pool bets are normalized so that each participant contributes a bet of one. Real pool payout schemes vary widely, although a simple scheme is to award all of the money to the player with the highest score, and in case of a tie to split the pot equally between the tied players. We call this the *standard payoff scheme*. This scheme is also a model for a winner-take-all pool with a tiebreaker that is reasonably independent from the game picks. For example, many football pools break ties with predictions about scoring in the Monday night game.

The techniques in this paper are applicable to a wide variety of payout schemes, but greater generality would introduce notational and computational complexity, which we chose to avoid. Instead, we assume throughout the paper that all pools use the standard payoff scheme.

The pool entrants consist of N *competitors* or *opponents* plus one distinguished player, for a total of $N+1$ participants. The standard payoff scheme means that players who tie for first split the $N+1$-sized pot equally.

2.2. Pool Probabilities

The fundamental assumption in this paper is that each opponent makes their picks randomly and independently for each game. To be more precise, for each matchup of two teams i and j, there is a number p_{ij} called the *pool probability* for that matchup. A given opponent picks the winner in the i versus j match by choosing team i with probability p_{ij} and team j with probability $1-p_{ij}$. An individual opponent picks each games' outcome independently of his picks for every other game. In addition, each opponent makes picks independently of every other opponent. The probabilities p_{ij} are fixed for the entire pool so that all opponents are making picks according to identical distributions.

For tournament pools we assume that opponents pick using a Markov process, where they first pick Round 1 winners to get Round 2 matchups, then independently pick the Round 2 winners, and so on to the champion. Although humans may not actually pick teams in this way, simulations presented in §5.5 show that picks made according to the model behave comparably to human-made picks.

Perhaps surprisingly, it is easy to find excellent data to use for pool probabilities. There are a number of large (over 100,000 player) free pools on the Internet, and some publish statistics on picks. For example, ESPN's Pigskin Pick'em gives the percentage of players choosing each football game for the week. For the NCAA tournament, Yahoo Tournament Pick'em has published the percentage of players picking each team to reach each round.

After the games begin, most online pools allow inspection of all participant picks. We were able to automatically retrieve 500,000 complete NCAA basketball poolsheets from ESPN's 2004 Tournament Challenge to use as sample data for retroactive analysis.

2.3. Actual Probabilities

The second main assumption of the model is that for each pair of teams i and j, there is a known *actual probability* a_{ij} that team i beats team j, and that the results of one game are independent of other games, and independent of earlier round games (in the context of elimination tournaments).

To estimate actual probabilities, there are many alternatives. There are a number of computer models on the Internet, such as the Sagarin rankings (Sagarin 2004) and Massey rankings (Massey 2004). There are also various approximations to the NCAA basketball RPI, the official rating system used by the NCAA when seeding teams in the tournament. Statistical models such as Bradley and Terry (1952), Boulier and Stekler (1999), Caudill (2003), and Kvam and Sokol (2006) attempt to predict outcomes in basketball tournaments. One could also derive data from "Las Vegas" odds. In Kaplan and Garstka (2001), there is a detailed discussion of possibilities in the context of the NCAA basketball tournament.

It might also be possible to use pool probabilities to derive the actual probabilities based on empirical data from past pools. Yet another alternative for the NCAA basketball tournament is to use the historical performance of seeds. The accuracy of seeding as a predictor is examined in Caudill and Godwin (2002).

There are really two issues here: finding accurate a_{ij}, and making the best use of that knowledge. This paper is only concerned with the second problem, and will give optimal results if the a_{ij} really are the actual probabilities of the games. On the other hand, one wants methods that are relatively stable. We give some evidence in §§3.5 and 5.6 that the methods in this paper will generate reasonable picks for a variety of a_{ij}.

2.4. The Optimization Problem

The goal of the remainder of the paper is to understand which picks maximize the expected return on a bet. The inputs to the problem are:

- N, the number of competitors in the pool.
- Actual probabilities a_{ij} that govern the outcomes of the games.
- Pool probabilities p_{ij} that describe behavior of the competitors in the pool.

Together with the assumptions in the previous three sections, these data describe a model sports betting pool where both the picks of all competitors and the outcome of the games are random variables. These variables range over the set $\mathcal{O}$ of all possible outcomes of the games.

The set $\mathcal{O}$ is finite—for example, in a 16-game football pool, $|\mathcal{O}| = 2^{16}$. There is one random variable $\mathbf{x}_\alpha \in \mathcal{O}$ for each competitor ($\alpha = 1 \ldots N$) giving that competitor's picks. There is one random variable $\mathbf{v} \in \mathcal{O}$ that represents the outcome of the games.

Now fix $\mathbf{y} \in \mathcal{O}$. A pool consists of $N+1$ players: N competitors who each bet one on their picks $\mathbf{x}_\alpha$ and one distinguished player who bets one on $\mathbf{y}$. The outcome $\mathbf{v}$ then determines a winner or group of winners of the pool, and they split the $N+1$-sized pot. The share of the pot ($\in[0, N+1]$) received by the distinguished player is the *return* from betting one on $\mathbf{y}$. Note that the *value* of a bet is the return minus the bet amount. In this paper, we will always use return instead of value because bets will always be one, and including the bet cost in every formula would lead to a worthless abundance of -1s.

The decision variable for the problem is $\mathbf{y} \in \mathcal{O}$. More traditionally, one could think of $\mathbf{y}$ as a collection of $\{0, 1\}$-valued decision variables, one for each game. Because the game outcomes and opponent picks are random, the return from betting one on $\mathbf{y}$ is random. To avoid extra notation, we simply write $E(\mathbf{y})$ for the expected value of the return from betting one on $\mathbf{y}$. The optimization problem is to find $\mathbf{y}$ that maximizes $E(\mathbf{y})$.

3. Football Pools

A typical office football pool covers one weekend of NFL football, which consists of 14–16 games. Before the weekend, each player chooses a winner for each of the games. When the games are finished, players are scored based on their number of correct predictions. We use the term *football pool* for any pool that requires players to make predictions for multiple independent two-outcome events.

The pool consists of g games, and in each game one team is arbitrarily designated the "favorite," whereas the other team is known as the "underdog." The favorite in game i has probability $a_i \in [0, 1]$ of winning, which we will call the *actual probability* for game i. Each of your opponents bets by choosing the favorite in game i with probability $p_i \in [0, 1]$, which we call the *pool probability* for game i.

The terms "favorite" and "underdog" should be read with care because the values of a_i and p_i may not both lie on the same side of 0.5. Generally, we designate the favorite so that $a_i \geqslant 0.5$, and use the terms "actual favorite" and "pool favorite" when needed for clarity.

3.1. Examples

We give some special cases and examples to illustrate the complexity of the expected return optimization problem.

Example 3.1 (One-Game Pools). Suppose that $g = 1$, and let $a = a_1 \geqslant \frac{1}{2}$ and $p = p_1$. Against one opponent, betting the actual favorite returns $p + 2a(1-p)$, and betting the underdog returns $(1-p) + 2(1-a)p$. Thus, one should bet the actual favorite when $N = 1$.

Now consider the limit as $N \to \infty$, and assume that $p \neq 0, 1$. Because the probability goes to one that there will be opponents betting on each team, the return on a bet is zero unless the pick is correct. Betting the favorite is correct with probability a, and splits the $N+1$-sized pot with Np opponents. As $N \to \infty$, the expected return is then a/p. Similarly, the expected return for an underdog bet is $(1-a)/(1-p)$. From this, we see that the favorite is better when $a > p$ and the underdog is better when $a < p$. We call this strategy *betting the edge*.

In general, for a one-game pool with N competitors, the threshold between betting the favorite and betting the underdog is given by

$$a = \left(1 + \left(\frac{1-p}{p}\right)\left(\frac{1-(1-p)^N}{1-p^N}\right)\right)^{-1}, \tag{3.1}$$

which interpolates between $a = \frac{1}{2}$ and $a = p$ as shown in Figure 1 for $N = 1, \ldots, 15$.

One might hope to understand a multigame pool as a collection of unrelated one-game pools. However, things are not so simple. Tricky parity issues can enter the picture when making picks. For example, there are times one should bet the actual underdog even when all opponents are picking the underdog as well:

EXAMPLE 3.2. Consider a pool with two games, with both actual probabilities just a bit more than 0.5, and both pool probabilities close to zero. The actual favorites are then FF, and UU is what everyone is betting. For a bet of FF, the probability of getting both games right and winning the pool is about 0.25. The probability of getting one game right and tying with everyone is about 0.5. The expected return is about $N/4$. However, a bet of FU or UF is sure to tie on the U game and has a just better than 50% shot at the F game, giving a return of about $N/2$.

To get a feel for the complexity of the picking problem, consider Figure 2. These pictures describe optimal strategies for pools with equivalent games, that is, pools where $a_i = a$ and $p_i = p$ for all i. The values of a and p vary along the axes, and the point at (a, p) is colored for the optimal number of favorites to pick (with black meaning all underdogs and white meaning all favorites).

With four games and three competitors, we see the parity issue of Example 3.2 as a tail of gray extending above the

Figure 1. Thresholds for the one-game pool.

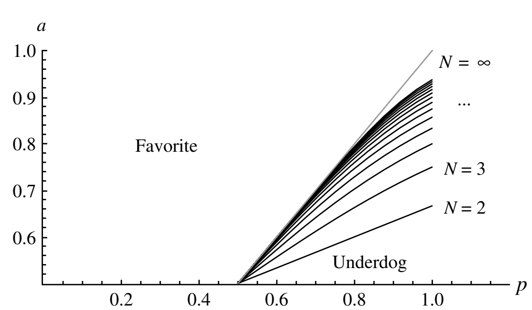
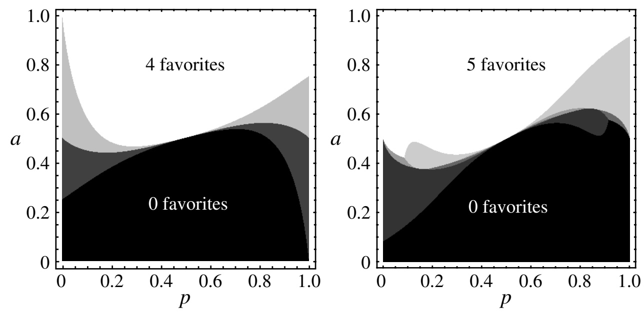


Figure 2. Equivalent games.



4 games, 3 competitors   5 games, 11 competitors

$a = 0.5$ line. In that region, the best pick is three instead of four favorites for parity reasons. With five games and 11 competitors, all six possible bets do show up as optimal for some values of a and p, and the complicated geometry of the regions is apparent.

3.2. Expected Return

In this section, we compute an exact formula for $E(\mathbf{y})$, the expected value of the return from betting one on picks $\mathbf{y}$.

We need some notation. For picks (or outcomes) $\mathbf{x}$ and $\mathbf{y}$, let $\mathbf{x} \wedge \mathbf{y}$ be the number of games for which $\mathbf{x}$ and $\mathbf{y}$ agree. Given probabilities p_i for the g games, let $P(\mathbf{x})$ be the probability that one opponent picks $\mathbf{x}$ exactly. That is,

$$P(\mathbf{x}) = \prod_{i=1}^{g} \begin{cases} p_i & \text{if the favorite is picked to win game } i \text{ in } \mathbf{x}, \\ (1-p_i) & \text{if the underdog is picked to win game } i \text{ in } \mathbf{x}. \end{cases}$$

Similarly, $A(\mathbf{x})$ is the actual probability that outcome $\mathbf{x}$ occurs.

Summing over all possible outcomes $\mathcal{O}$ of the games, the expected return for a bet on $\mathbf{y}$ is

$$E(\mathbf{y}) = \sum_{\mathbf{x} \in \mathcal{O}} A(\mathbf{x}) E(\mathbf{y} \mid \mathbf{x}), \tag{3.2}$$

where $E(\mathbf{y} \mid \mathbf{x})$ is the expected return on $\mathbf{y}$ given the outcome $\mathbf{x}$. The quantity $E(\mathbf{y} \mid \mathbf{x})$ depends on $\mathbf{y}$ only as far as the score $s = \mathbf{x} \wedge \mathbf{y}$. Therefore, let $\mathcal{F}(\mathbf{x}, s)$ be the expected return, given an outcome of $\mathbf{x}$ and score s. Then,

$$E(\mathbf{y}) = \sum_{\mathbf{x} \in \mathcal{O}} A(\mathbf{x}) \mathcal{F}(\mathbf{x}, \mathbf{x} \wedge \mathbf{y}). \tag{3.3}$$

We now turn to the problem of computing $\mathcal{F}(\mathbf{x}, s)$. With the standard payoff assumption, the return is nonzero when s is the highest score or s is tied with some number of opponents. The first step in the computation is to study a single opponent.

Let $\mathcal{E}(\mathbf{x}, s)$ and $\mathcal{L}(\mathbf{x}, s)$ denote the conditional probability that a given opponent scores equal to s or less than s, given that outcome **x** actually occurred. These functions depend implicitly on the pool probabilities p_i.

Then, we have

$$\mathscr{E}(\mathbf{x}, s) = \sum_{\mathbf{z}\in\mathscr{O};\, \mathbf{z}\wedge\mathbf{x}=s} P(\mathbf{z}), \tag{3.4}$$

$$\mathscr{L}(\mathbf{x}, s) = \sum_{k=0}^{s-1} \mathscr{E}(\mathbf{x}, k) = \sum_{\mathbf{z}\in\mathscr{O};\, \mathbf{z}\wedge\mathbf{x}<s} P(\mathbf{z}). \tag{3.5}$$

To get an expression for $\mathscr{F}$, we compute the probability of tying with k competitors and beating the rest, then divide by $k+1$, the number of winners splitting the pot:

$$\mathscr{F}(\mathbf{x}, s) = \sum_{k=0}^{N} \frac{N+1}{k+1}\binom{N}{k}\mathscr{L}(\mathbf{x}, s)^{N-k}\mathscr{E}(\mathbf{x}, s)^k \tag{3.6}$$

$$= \sum_{k=0}^{N} \binom{N+1}{k+1}\mathscr{L}(\mathbf{x}, s)^{N-k}\mathscr{E}(\mathbf{x}, s)^k \tag{3.7}$$

$$= \frac{(\mathscr{L}(\mathbf{x}, s)+\mathscr{E}(\mathbf{x}, s))^{N+1} - \mathscr{L}(\mathbf{x}, s)^{N+1}}{\mathscr{E}(\mathbf{x}, s)}. \tag{3.8}$$

The last equality follows from the binomial formula.

Putting (3.8) together with (3.3) proves the following:

Theorem 3.3. *In a football pool with N competitors and the standard payoff scheme, the expected return for a bet on games* **y** *is*

$$E(\mathbf{y}) = \sum_{\mathbf{x}\in\mathscr{O}} A(\mathbf{x}) \frac{(\mathscr{L}(\mathbf{x},\mathbf{x}\wedge\mathbf{y})+\mathscr{E}(\mathbf{x},\mathbf{x}\wedge\mathbf{y}))^{N+1} - \mathscr{L}(\mathbf{x},\mathbf{x}\wedge\mathbf{y})^{N+1}}{\mathscr{E}(\mathbf{x},\mathbf{x}\wedge\mathbf{y})}. \tag{3.9}$$

We saw in Example 3.1 that the optimal strategy for a one-game pool interpolates between betting the actual favorites and betting the edge, as N goes from 1 to ∞. For multigame pools, even the one-opponent case is difficult. However, we have the following:

Proposition 3.4. *For a g-game football pool, let* $\mathbf{e} \in \mathscr{O}$ *pick the edge in every game. That is,* **e** *picks the favorite in game i when $a_i \geqslant p_i$ and the underdog when $a_i < p_i$. Then, for any picks* $\mathbf{y} \in \mathscr{O}$, $\lim_{N\to\infty} E(\mathbf{e}) \geqslant \lim_{N\to\infty} E(\mathbf{y})$.

Proof. For any **y**, $\lim_{N\to\infty} E(\mathbf{y}) = A(\mathbf{y})/P(\mathbf{y})$ because **y** has probability $A(\mathbf{y})$ of being perfect and thus splitting the $N+1$-sized pot with $N \cdot P(\mathbf{y})$ opponents. The quantity $A(\mathbf{y})/P(\mathbf{y})$ is maximal for $\mathbf{y} = \mathbf{e}$. □

Computing $E(\mathbf{y})$ for a bet **y** requires exponential time. More precisely, the expression in (3.9) has $O(4^g)$ terms. A technical improvement described in Appendix A.1 reduces the number of computations to $O(2^g)$, without which an NFL football pool would be intractable. In §4, we describe a technique to compute an approximation of $E(\mathbf{y})$ quickly, which must be used for pools with a large number of games.

For NFL football pools, it is reasonable to perform an exhaustive search of all possible bets Y. However, one could also apply standard search techniques, such as greedy or genetic algorithms. As a concrete example, declare two bets **y** and **y**′ to be k-neighbors if they differ in exactly k games. In our experiments, a hill-climbing search terminating at a point maximal among its 2-neighbors has never failed to find the best picks.

3.3. The 2004–2005 NFL Season

We tested our methodology during the 2004–2005 NFL season by entering 4–6 free online pools per week, with the number of competitors N varying from approximately 400 to approximately 200,000. All pools broke ties using the Monday night football score in some form or another. These pools were free, but the model in this paper still applies to find optimal bets—the payoffs are simply scaled. In particular, the ESPN Pigskin Pick'em pool, with 170,000 competitors, was offering a fine ESPN logo hat as their weekly prize.

ESPN's pool was our primary source of pool probabilities p_i. Each Tuesday, ESPN released the percentages of competitors picking each game of the next week's pool, and then updated this information over the course of the week.

For actual probabilities a_i, we used computer-generated predictions made weekly by Massey (2004). Massey ratings have a published algorithm that is mathematically straightforward and depends only on game scores, venues, and dates.

Table 1 summarizes the Massey-predicted probabilities a_i and the ESPN competitor percentages p_i. Each row shows the number of games that had a_i (respectively, p_i) in a given range, and the percentage of those games that had the predicted result. ESPN competitors had a slightly better record overall than the Massey computer model, but the most striking feature of Table 1 is that more than half the games had a favorite that attracted over 80% of the competitors.

Table 1. Actual and pool probabilities for the NFL 2004–2005 season.

	Massey actual		ESPN pool	
Prediction (%)	No. of games	Correct (%)	No. of games	Correct (%)
50–55	53	56.6	14	50.0
55–60	41	48.8	14	42.9
60–65	50	64.0	23	39.1
65–70	35	62.9	21	66.7
70–75	35	74.3	24	70.8
75–80	27	66.7	23	56.5
80–85	6	100.0	36	69.4
85–90	7	85.7	32	53.1
90–95	2	50.0	35	77.1
95–100	0	—	34	82.4
Totals	256	62.9	256	63.7

Table 2. Examples of NFL football picks.

NFL 2004–2005 Week 9				NFL 2004–2005 Week 14			
Pool favorite	Pool underdog	ESPN p	Massey a	Pool favorite	Pool underdog	ESPN p	Massey a
NYG	**CHI***	0.968	0.72	**DEN**	MIA*	0.977	0.75
SEA	SF*	0.949	0.58	**IND**	HOU*	0.961	0.72
NYJ	**BUF***	0.914	0.57	**GB**	DET*	0.957	0.74
SD*	NO	0.900	0.76	**BUF***	CLE	0.954	0.75
KC	**TB***	0.891	0.48	**NE***	CIN	0.951	0.87
DET	**WAS***	0.882	0.65	**PHI***	WAS	0.944	0.77
BAL*	CLE	0.885	0.74	**ATL**	OAK*	0.932	0.66
IND*	MIN	0.792	0.69	**BAL***	NYG	0.930	0.79
DEN*	HOU	0.786	0.64	ARZ	**SF***	0.899	0.65
CAR*	**OAK**	0.759	0.65	**DAL**	NO*	0.891	0.63
NE	STL*	0.750	0.61	**JAX**	CHI*	0.851	0.65
DAL	**CIN***	0.557	0.43	**SD***	TB	0.848	0.68
MIA*	**ARZ**	0.513	0.66	MIN*	**SEA**	0.843	0.66
PIT	PHI*	0.541	0.42	**PIT***	NYJ	0.804	0.65
				CAR*	STL	0.690	0.64
				KC	TEN*	0.590	0.47
Upsets (pool opinion): 7				Upsets (pool opinion): 2			
Correct picks: 9				Correct picks: 8			

Notes. Actual game winners are bold. Optimal picks for $N = 170{,}000$ are starred*.

Each week, using $N = 170{,}000$, we computed $E(\mathbf{y})$ for every possible bet $\mathbf{y}$. The picks $\mathbf{b}$ that maximized $E(\mathbf{b})$ will be called the "optimal picks." Because the optimal picks depend on N, we had to repeat the search for various N to enter each week's collection of pools. With our implementation, searching all possible bets for a given N took about 10 minutes for 14-game weeks and about four hours for 16-game weeks. To save time, we used the hill-climbing search described above for values of $N \neq 170{,}000$.

Two example weeks from 2004 are shown in Table 2. The first two columns show the teams in each game, with the ESPN pool favorite on the left. The optimal picks (for $N = 170{,}000$) are starred, and the actual game winners are shown in boldface. The two numeric columns are the percentage of ESPN competitors picking the pool favorite and Massey's prediction for the probability of the pool favorite winning.

In Week 9, there were big upsets, and in all pools both the average and winning scores were low. The optimal picks in the 9,000-person CBS Sportsline pool scored 10 out of 14 (these picks were the same as those in Table 2 except with NE picked over STL). This was our best week, and in the CBS pool we finished two points behind the winner, tied for 19th place. Week 14 had few upsets, and the optimal picks did poorly although average and winning scores were high.

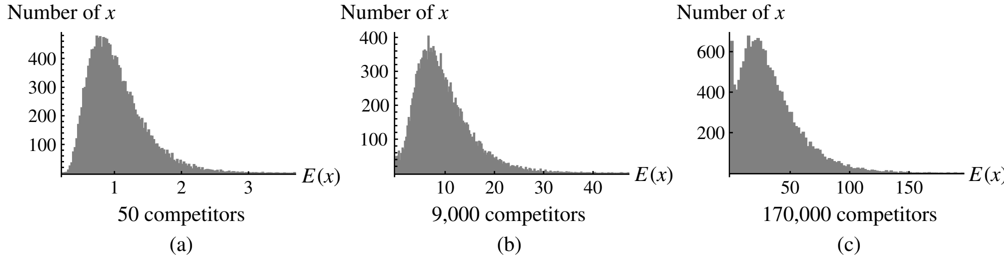
Table 3 summarizes weekly results for the 170,000-competitor ESPN Pigskin Pick'em pool. The main differences for smaller pools were relatively lower scores for winners, and lower estimates of expected return. For $N = 9{,}000$, the

Table 3. Summary of results for the NFL 2004–2005 season.

Week	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
Games	16	16	14	14	14	14	14	14	14	14	16	16	16	16	16	16	16
Correct picks[a]	9	8	3	3	7	4	5	9	9	8	6	5	9	9	6	7	8
Opp. average[b]	8.5	8.7	8.7	7.6	7.7	9.2	7.1	7.5	7	8	11.2	10.2	9.3	10.8	8.9	9.7	8
Exp. return[c]	209	226	83	331	71	118	195	86	294	55	392	115	186	142	193	58	134
Upset picks[d]	9	11	7	8	10	9	9	9	8	10	12	13	8	8	11	10	10
Upsets[e]	6	5	4	7	7	3	6	8	7	6	3	4	5	3	6	4	8
Winner[f]	16	15	14	14	14	14	13	14	13	14	16	15	15	16	16	16	16
No. Tied[g]	1	10	36	3	3	$\geqslant 50$	2	2	3	8	$\geqslant 50$	$\geqslant 50$	10	17	1	3	9

[a]Maximum return picks $\mathbf{b}$ with a_i from Massey, p_i from ESPN, and $N = 170{,}000$.
[b]Average score of ESPN Pigskin Pick'em participant.
[c]Calculated expected return $E(\mathbf{b})$ of the week's picks.
[d]Picks in $\mathbf{b}$ against the pool favorites.
[e]Upsets that actually occurred, according to pool favorites.
[f]Winning score for ESPN Pigskin Pick'em.
[g]Players tied with winning score.

Figure 3. Distribution of expected return values for all 16,384 possible picks for Week 7.



expected return for optimal picks ranged from 16 to 74 with an average of 41 over all 17 weeks. For $N = 50$, expected return ranged from 2.26 to 7.17, with an average of 3.87. Unfortunately, with only 17 weeks per year for testing, one needs to wait quite a while for expected returns to pay off. For example, one only needs to win a 9,000-person pool about once every 15–20 years to achieve a return of 40.

It should be clear from the data that the purpose of the optimal picks is not to get a high score from week to week. In fact, the $N = 170{,}000$ optimized picks went against Massey recommendations in 44% of the games, meaning they picked an average of 6.6 actual underdogs per week, ranging from a low of 2 to a high of 8. The picks went against the ESPN pool favorites 63% of the time, an average of 9.5 games per week.

3.4. Picking Strategies

For a comparison of various picking strategies, consider Figure 3. Each picture is a frequency distribution of expected returns for all of the 16,384 possible bets in Week 7, first for $N = 50$, then $N = 9{,}000$, and then $N = 170{,}000$. The surprising thing about these distributions is that most of the 16,384 possible picks are good, meaning $E(\mathbf{x}) > 1$ for most bets $\mathbf{x}$. The pool is a zero-sum game, so this indicates that almost every opponent is making one of the small number of bad picks. To double check this remarkable conclusion, we gathered 24,000 poolsheets from ESPN's Pigskin Pick'em for Week 7. We found that the top 25 most popular picks (which included all picks chosen by at least 0.5% of the participants) accounted for 62% of the sample.

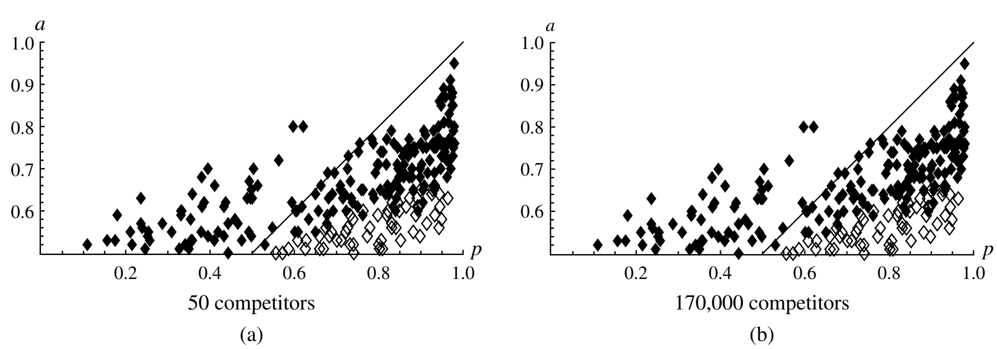
The upshot is that once football pools get large, crowd avoidance is crucial—even picking at random is better than picking lots of favorites. To make this more precise, we calculated the expected return for all possible picks over the entire season, and found that randomly selected picks had an average weekly expected return of 23.4 when $N = 170{,}000$. The optimal picks $\mathbf{b}$ varied from 3.4–28 times better than random over the 17-week season, plenty of reason to search for them.

In practice, finding good picks may not require much sophistication. The theory is complicated because optimal picks cannot be made on a game-by-game basis. That is, values of a_i and p_i do not by themselves determine the best pick in game i (Example 3.2 and Figure 2). However, Figure 4 shows that this rarely matters in practice.

Figure 4 summarizes the 2004-2005 NFL season optimal picks for pools with 170,000 and 50 competitors. There is one dot for each of the season's 256 games. Each dot is positioned at the (p, a) coordinates for the corresponding game, and is shown black when the optimal picks chose the actual favorite and is shown white for the underdog.

The striking feature of these figures is the nearly clean separation into a favorite zone and an underdog zone. Using these charts, one could make a decent set of picks by plotting the weeks' games on the chart and picking by zone on a game-by-game basis. It would be interesting to find a theoretical (or even empirical) formula for the apparent separating curve for general N.

Figure 4. 2004 NFL picks.



3.5. Sensitivity to Input Data

The optimal picks are not sensitive to small changes in N. Figure 4 gives an indication of how large changes in N affect the picks: There is a gradual switching of some picks from favorites to underdogs as N grows.

To test dependence on a_i, we repeatedly chose new values $\tilde{a}_i$ uniformly randomly from the interval $[a_i-0.01, a_i+0.01]$. We then took the optimal picks $\mathbf{b}$ for each week and computed $\tilde{E}(\mathbf{b})$ using $\{\tilde{a}_i\}_{i=1}^{g}$ as actual probabilities. The resulting values of $\tilde{E}(\mathbf{b})$ were approximately normally distributed with mean $E(\mathbf{b})$. The average weekly coefficient of variance for $\tilde{E}(\mathbf{b})$ was 2.1% for $N=50$, 3.5% for $N=9{,}000$, and 4.2% for $N=170{,}000$. Changing to $\tilde{a}_i\in[a_i-0.05, a_i+0.05]$ resulted in a five-fold increase in coefficient of variance, almost exactly.

Running the same test with p_i using $\tilde{p}_i\in[p_i-0.01, p_i+0.01]$ gave 1.2% for $N=50$, 3.5% for $N=9{,}000$, and 5.4% for $N=170{,}000$. The same five-fold increase was observed when using $p_i\pm 0.05$. Note that in small pools, variations in a_i have more impact, whereas in large pools, variations in p_i make a bigger difference. This reinforces the philosophy that small pools are about accurate predictions, whereas large pools are about crowd avoidance.

Values of a_i could easily be off by more than $\pm 5\%$, and therefore computed values of expected return should be read with some caution. On the other hand, the relative quality of picks appears to be more stable than the actual value of E. Figure 4 suggests that in practice, values of (a_i, p_i) away from the favorite/underdog division do not need to be particularly accurate. As a further test, we searched for optimal picks $\tilde{\mathbf{b}}$ using input data perturbed by up to ± 0.05 as above. We then calculated the position of $\tilde{\mathbf{b}}$ among all 16,384 picks ranked using the original input data. Repeating this test 20 times for each of the 14-game weeks, the ranks varied from 1 to 4,412 (out of 16,384), with a median of 6.

4. Normal Approximation

This section describes an approximation E^{norm} to the expected return on a bet, using the assumption that player scores are random variables with normal distributions.

4.1. Expected Return

Let the random variables $\{X_\alpha\}_{\alpha=1}^{N}$ be the scores of the N competitors in the pool, and let the random variable Y be the score for a fixed set of picks $\mathbf{y}$.

A player's score is a sum of scores for the individual games in the pool. These individual game scores have binomial distributions, but in a pool with sufficiently many games, the central limit theorem implies that their sum is approximately normally distributed. In this section, we assume that X_α and Y are normally distributed and derive a method for evaluating the quality of the picks $\mathbf{y}$.

Because each opponent is assumed to follow the same strategy, the mean and variance of X_α and of X_β coincide for all $\alpha, \beta\in 1, \ldots, N$.

Define the random variables $W_\alpha = X_\alpha - Y$, $\alpha = 1, \ldots, N$. The idea is that for the bet $\mathbf{y}$ to win anything, all the W_αs must be nonpositive. Let

$$\mu=\mu(W_\alpha)=\mu(X_\alpha)-\mu(Y), \tag{4.1}$$

$$\sigma^2=\sigma^2(W_\alpha)=\sigma^2(X_\alpha)+\sigma^2(Y)-2\operatorname{cov}(X_\alpha, Y), \tag{4.2}$$

$$\begin{aligned} c&=\operatorname{cov}(W_\alpha, W_\beta)\\ &=\operatorname{cov}(X_\alpha, X_\beta)+\sigma^2(Y)-2\operatorname{cov}(X_\alpha, Y). \end{aligned} \tag{4.3}$$

The next theorem measures the quality of the bet $\mathbf{y}$ entirely in terms of μ, σ^2, and c. This means that with the normality assumption, all of the pool information about $\mathbf{y}$, opponent perceptions, and actual probabilities of games boil down to just three numbers! The computation of μ, σ^2, and c from $\mathbf{y}$ and the pool data is done in §4.2 for football pools and in §A.2 for elimination tournaments. For now, we assume that μ, σ^2, and c are known.

Theorem 4.1. *For a fixed set of picks* $\mathbf{y}$, *put* μ, σ^2, *and* c *as above. Let*

$$\nu_m(t)=\frac{m-\mu-\sqrt{c}\,t}{\sqrt{\sigma^2-c}}. \tag{4.4}$$

The probability that picks $\mathbf{y}$ *will bet the sole winner in a pool with* N *opponents is approximately*

$$\text{Prob}(\mathbf{y}\textit{ is the sole winner}) =\int_{-\infty}^{\infty}\Phi(\nu_{-0.5}(t))^N\varphi(t)\,dt. \tag{4.5}$$

In a pool with the standard payoff scheme, the expected return on a bet of 1 *is approximately*

$$E^{norm}(\mathbf{y})=\int_{-\infty}^{\infty}\left[\frac{\Phi(\nu_{0.5}(t))^{N+1}-\Phi(\nu_{-0.5}(t))^{N+1}}{\Phi(\nu_{0.5}(t))-\Phi(\nu_{-0.5}(t))}\right]\varphi(t)\,dt. \tag{4.6}$$

Here $\varphi(t)=(2\pi)^{-1/2}e^{-t^2/2}$ *is the probability distribution function* (*p.d.f.*) *for a standard normal random variable, and* $\Phi(t)=\frac{1}{2}(1+\operatorname{erf}(t/\sqrt{2}))$ *is the associated cumulative distribution function* (*c.d.f.*).

Remark. All the 0.5s in Theorem 4.1 come from continuity corrections.

Proof. Following Dunnett and Sobel (1955), we let $Z_1, \ldots, Z_N, T$ be independent standard normal random variables, and write

$$W_\alpha=\sqrt{\sigma^2-c}\,Z_\alpha+\sqrt{c}\,T+\mu \tag{4.7}$$

for $\alpha=1, \ldots, N$. Now,

$$W_\alpha\leqslant m \iff Z_\alpha\leqslant\frac{m-\mu-\sqrt{c}\,T}{\sqrt{\sigma^2-c}}=\nu_m(T), \tag{4.8}$$

and we can compute the probability:

$$\mathrm{Prob}(\forall\,\alpha\colon W_\alpha \leqslant m) = \int_{-\infty}^{\infty}\int_{-\infty}^{\nu_m(t)}\cdots\int_{-\infty}^{\nu_m(t)} \varphi(z_1)\cdots\varphi(z_N)\varphi(t)\,dz_1\cdots dz_N\,dt \tag{4.9}$$

$$= \int_{-\infty}^{\infty}\Phi(\nu_m(t))^N\varphi(t)\,dt. \tag{4.10}$$

The probability that picks $\mathbf{y}$ win the pool outright is $\mathrm{Prob}(\forall\,\alpha\colon W_\alpha \leqslant -0.5)$ (where -0.5 is a continuity correction to zero), and this establishes (4.5).

More generally, the probability of tying with the first k competitors and beating the rest is given by

$$Q_k = \mathrm{Prob}(W_\alpha \in [-0.5, 0.5], \alpha = 1\ldots k;\quad W_\alpha \leqslant -0.5, \alpha = k+1\ldots N) \tag{4.11}$$

$$= \int_{-\infty}^{\infty}\int_{-\infty}^{\nu_{-0.5}(t)} \overset{N-k}{\cdots} \int_{-\infty}^{\nu_{-0.5}(t)}\int_{\nu_{-0.5}(t)}^{\nu_{0.5}(t)} \overset{k}{\cdots} \int_{\nu_{-0.5}(t)}^{\nu_{0.5}(t)} \varphi(z_1)\cdots\varphi(z_N)\varphi(t)\,dz_1\cdots dz_N\,dt \tag{4.12}$$

$$= \int_{-\infty}^{\infty}[(\Phi(\nu_{0.5}(t)) - \Phi(\nu_{-0.5}(t)))^k \cdot \Phi(\nu_{-0.5}(t))^{N-k}]\varphi(t)\,dt. \tag{4.13}$$

The expected return on a bet of one with picks $\mathbf{y}$ is then

$$E^{\mathrm{norm}}(\mathbf{y}) = \sum_{k=0}^{N}\frac{N+1}{k+1}\binom{N}{k}Q_k$$
$$= \int_{-\infty}^{\infty}\left[\sum_{k=0}^{N}\frac{N+1}{k+1}\binom{N}{k}(\Phi(\nu_{0.5}(t)) - \Phi(\nu_{-0.5}(t)))^k \cdot \Phi(\nu_{-0.5}(t))^{N-k}\right]\varphi(t)\,dt$$
$$= \int_{-\infty}^{\infty}\left[\frac{\Phi(\nu_{0.5}(t))^{N+1} - \Phi(\nu_{-0.5}(t))^{N+1}}{\Phi(\nu_{0.5}(t)) - \Phi(\nu_{-0.5}(t))}\right]\varphi(t)\,dt.$$

The final step used the binomial formula in the same manner as (3.8). □

4.2. Normal Approximation Applied to Football Pools

To apply Theorem 4.1 to football pools, we need to compute μ, σ^2, and c for any set of picks $\mathbf{y}$. Equations (4.1)–(4.3) reduce the problem to the following:

Proposition 4.2. *Suppose that one player makes one set of picks using probabilities $\{p_i\}$, and has a score given by the random variable X. Then, the mean and variance of X are*

$$\mu(X) = \sum_{i=1}^{g} a_i p_i + (1-a_i)(1-p_i), \tag{4.14}$$

$$\sigma^2(X) = \sum_{i=1}^{g}(a_i + p_i - 2a_ip_i)(1 - a_i - p_i + 2a_ip_i). \tag{4.15}$$

If a second player makes one set of picks using probabilities $\{q_i\}$, and has a score given by Y, then

$$\mathrm{cov}(X, Y) = \sum_{i=1}^{g} 4a_i(1-a_i)\left(p_i - \frac{1}{2}\right)\left(q_i - \frac{1}{2}\right). \tag{4.16}$$

To compute the covariance between two opponent scores, specialize to $q_i = p_i$. To evaluate a fixed bet, let p_i (or q_i) $\in \{0, 1\}$.

Proof. Both X_α and Y can be written as sums of random variables, one for each game of the pool. Because the summands are independent, μ, σ^2, and cov distribute over the sums and the problem reduces to the one game case, which is straightforward. □

Evaluating the approximate expected return $E^{\mathrm{norm}}(\mathbf{y})$ for a given set of picks $\mathbf{y}$ is now easy. Compute μ, σ^2, and c for the given $\mathbf{y}$ and then perform the numeric integration (4.6). The process is fast enough that our implementation can evaluate all 65,536 bets for a 16-game pool in a few seconds.

As a test of the normal approximation method, we used it to find the best (highest E^{norm}) picks $\mathbf{b}_1^{\mathrm{norm}}, \ldots, \mathbf{b}_{17}^{\mathrm{norm}}$ for each week of the 2004-2005 NFL season in a 170,000-person pool. The approximated return $E^{\mathrm{norm}}(\mathbf{b}_i^{\mathrm{norm}})$ was off by an average of 24% from the exact return $E(\mathbf{b}_i^{\mathrm{norm}})$, ranging from 43% too low to 48% too high. While this is not encouraging, at least the order of magnitude is correct.

Happily, it appears that relative quality of picks is roughly preserved when using normal approximation. For each week, we used the exact formula of §3.2 to find the rank of $E(\mathbf{b}_i^{\mathrm{norm}})$ among all bets. This is the first row of data in Table 4. In 11 of 17 weeks, the pick maximizing E^{norm} was in the top ten for E. The second row of Table 4 shows $E(\mathbf{b}_i^{\mathrm{norm}})$ as a percentage of the maximum possible E for the week. Using the normal approximation always found good picks and often found excellent picks. In the next section, we are forced to use the normal approximation exclusively.

Table 4. Best picks found with normal approximation.

Week	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
Rank	2	1	3	1	9	18	4	1	2	170	3	4	289	88	1	91	15
% of best	100	100	87	100	92	75	94	100	98	70	96	95	71	81	100	84	88

5. Elimination Tournaments

In a single-elimination tournament with R rounds, there are 2^R teams. Before the tournament begins, the teams are placed into a "bracket." Each round, the teams play according to the bracket, the losers are eliminated, and the winners advance to the next round. This system is common in two-player/two-team sports, including major sport playoffs, most tennis tournaments, and the NCAA basketball tournament, which is our motivating example.

A tournament pool is often run as follows: Each competitor predicts in advance how the entire tournament will play out, and then scores points for each correct pick. Correctly picking a team to reach a later round is generally worth more, and we write w_r for the value of a correct pick in round r. A common sequence is $w_r = 2^r$, which gives each round the same total point value.

As a concrete example, the ESPN Men's Tournament Challenge is a free, nationwide pool for the NCAA men's basketball tournament. The tournament has six rounds (so 64 teams), and the ESPN pool had approximately five million competitors in 2004. Correct picks score 10, 20, 40, 80, 120, and 160 points in rounds 1–6, and the 2004 winner had 1,330 out of a possible 1,680 points.

As before, we assume knowledge of both the actual probability of events as well as information about opponent picks. Note that any given pair of teams can play each other over the course of the tournament. For each pair of teams (i, j) in the tournament, assume that we know the actual probability a_{ij} that team i beats team j. This information could potentially be generated by a computer model.

Also assume that we know the probability p_{ij} that an opponent will pick team i to beat team j. Note that this information is rarely available, in part because it is unlikely to be published by pool organizers, but more so because many potential matchups in a large tournament will not appear in a significant number of pool entries. We return to this issue in §5.2.

The problem is to optimize $E(\mathbf{y})$ over $\mathbf{y} \in \mathcal{O}$, where $\mathcal{O}$ is the set of all possible outcomes of the tournament. Because there are $2^R - 1$ games, $|\mathcal{O}| = 2^{2^R-1}$.

For an outcome $\mathbf{x} \in \mathcal{O}$, define $A(\mathbf{x})$ to be the probability that $\mathbf{x}$ occurs given the collection of head-to-head probabilities $\{a_{ij}\}$. Here,

$$A(\mathbf{x}) = \Pi_{i,j} a_{ij},$$

where the product runs over all $2^R - 1$ pairs (i, j) where team i plays and beats team j in the bracket $\mathbf{x}$. For an event $U \subset \mathcal{O}$, define $A(U)$ to be the probability that U occurs, given $\{a_{ij}\}$. This is simply the sum $\sum_{\mathbf{x}\in U} A(\mathbf{x})$. We similarly define $P(U)$ associated to $\{p_{ij}\}$.

Define the event "$i \to r$" $\subset \mathcal{O}$ to be the set of outcomes where team i has reached and won its round r game. Then, $A(i \to r)$ is the actual probability that team i wins round r, and $P(i \to r)$ is the percentage of opponents that picked team i to win round r.

5.1. Canonical Picks

For football pools, there are two natural sets of picks—picking all favorites and picking the edge in every game. "All favorites" is the most likely outcome and maximizes expected score. "The edge" maximizes expected return for large N (Proposition 3.4).

This section discusses analogous canonical picks for tournament pools. In addition, we introduce a fundamental induction technique due to Kaplan and Garstka (2001) for computations.

The following example shows that tournaments do introduce some complications:

Example 5.1. A four-team tournament with teams A, B, C, and D pits A versus B and C versus D in Round 1, with the winners meeting for the final. Assume that A always beats B, and C beats D with probability 0.6. Finally, A always beats D, but has only 0.5 probability of beating C. The only possible outcomes are: A wins over C (probability 0.3), C wins over A (probability 0.3), and A wins over D (probability 0.4).

We see that the most likely outcome contains the upset D beats C.

A tournament bracket $\mathbf{x}$ consists of two halves—the *top half bracket* $\mathbf{x}_{\text{top}}$ and the *bottom half bracket* $\mathbf{x}_{\text{bot}}$, with one team from each half reaching the final game.

To optimize some quantity of a bracket, we follow an inductive procedure. The inductive hypothesis is that we know, for each team i, the optimal half-bracket with team i winning. In the inductive step, we must compute each team's optimal whole bracket from the half-bracket information. The idea is to optimize over all possible final round opponents for that team, and then fill in the rest of the bracket using the optimal half-brackets for the two finalists. The actual details depend on the quantity to be optimized, and we give three examples below. These computations are of polynomial complexity in the number of teams.

Example 5.2 (Most-Likely Bracket). The most-likely bracket maximizes $A(\mathbf{x})$ over brackets $\mathbf{x} \in \mathcal{O}$. If team i beats team j in the final of bracket $\mathbf{x}$,

$$A(\mathbf{x}) = A(\mathbf{x}_{\text{top}}) \cdot a_{ij} \cdot A(\mathbf{x}_{\text{bot}}). \tag{5.1}$$

Our inductive hypothesis means we know the optimal choice of $\mathbf{x}_{\text{top}}$ and $\mathbf{x}_{\text{bot}}$ for fixed i and j. To compute the optimal $\mathbf{x}$ with team i winning, we maximize the value (5.1) over all choices of j.

Example 5.3 (Very Large Pools). We want to find $\mathbf{x}$ that maximizes expected return $E(\mathbf{x})$ in the limit as the number N of competitors goes to ∞. As $N \to \infty$, every possible bracket is picked by some opponent. Then, picks $\mathbf{x}$ must be perfect to win a share of the pot, and this happens with probability $A(\mathbf{x})$. In this case, the pot will be split with

$N \cdot P(\mathbf{x})$ competitors. Then, $\lim_{N\to\infty} E(\mathbf{x}) = A(\mathbf{x})/P(\mathbf{x})$. If team i beats team j in the final of bracket $\mathbf{x}$,

$$\frac{A(\mathbf{x})}{P(\mathbf{x})} = \frac{A(\mathbf{x}_{\text{top}})}{P(\mathbf{x}_{\text{top}})} \cdot \frac{a_{ij}}{p_{ij}} \cdot \frac{A(\mathbf{x}_{\text{bot}})}{P(\mathbf{x}_{\text{bot}})}, \tag{5.2}$$

and we can proceed as in Example 5.2.

It is worth noting that "very large" for a six-round tournament is well in excess of 2^{63} competitors, so these picks are of little practical use. As an example, with data from the 2004 NCAA Men's Basketball Tournament, these limit picks had the 1, 11, 16, and 4 seeds (from the four regions) in the final four.

Example 5.4 (Maximum Expected Score). We want to find the bracket $\mathbf{b}_{\text{score}}$ that maximizes the expected score. Write $\mu_T(\cdot)$ for the expected total score of a bracket (or partial bracket). If team i is picked as the winner of a bracket $\mathbf{x}$, then

$$\mu_T(\mathbf{x}) = \mu_T(\mathbf{x}_{\text{top}}) + \mu_T(\mathbf{x}_{\text{bot}}) + w_r A(i \to r), \tag{5.3}$$

where $\mathbf{x}$ has r rounds. Assume without loss of generality that i comes from $\mathbf{x}_{\text{top}}$. The inductive hypothesis means that we know the optimal choice of $\mathbf{x}_{\text{top}}$ with i winning. The optimal $\mathbf{x}$ is found by maximizing over all possible winners of $\mathbf{x}_{\text{bot}}$. Finally, $\mathbf{b}_{\text{score}}$ is found by maximizing over all possible winners of $\mathbf{x}$. This recursion is due to Kaplan and Garstka (2001), where it is explained in detail. A worthy implementation is available at T. Adams' website (Adams 2004).

5.2. Expected Return

This section describes a method for evaluating the quality of picks $\mathbf{y}$ in terms of expected return on a bet of one. Calculating $E(\mathbf{y})$ exactly for even one $\mathbf{y}$ seems intractable, so we turn our attention to the computation of the normal approximation $E^{\text{norm}}(\mathbf{y})$. The basic assumptions are that opponent scores are modeled by normal random variables $\{X_\alpha\}_1^N$ and that the score of picks $\mathbf{y}$ is modeled by a normal random variable Y. Once the means, variances, and covariances of these variables are known, Theorem 4.1 computes $E^{\text{norm}}(\mathbf{y})$.

Formulas to compute the vital statistics of $X = X_\alpha$ and Y are given in the appendix in Proposition A.2 and Proposition A.3. A crucial feature of these formulas is that they are written in terms of event probabilities $A(-)$ and $P(-)$ rather than directly involving the head-to-head data $\{a_{ij}\}$ and $\{p_{ij}\}$. In particular, we need to know $A(i \to r)$, $A(i \to r \cap j \to s)$, $P(i \to r)$, and $P(i \to r \cap j \to s)$ for all teams i, j and rounds r, s.

Given $\{a_{ij}\}$, the probabilities $A(i \to r)$ and $A(i \to r \cap j \to s)$ can be computed with the induction technique used above. For the former,

$$A(i \to 0) = 1,$$

$$A(i \to r+1) = A(i \to r) \sum_k a_{ik} A(k \to r), \tag{5.4}$$

where k runs over all 2^{r-1} possible round r opponents of team i. For the latter, assume that $r \geqslant s$ and apply the following cases inductively:

$$A(i \to r \cap j \to s) = \begin{cases} A(i \to r) & \text{if } i = j, \\ 0 & \text{if } s \geqslant m, \\ A(i \to r) A(j \to s) & \text{if } r < m, \\ A(i \to r-1 \cap j \to s) \sum_k a_{ik} A(k \to r-1) & \text{if } r > m > s, \\ A(i \to r-1) \sum_k a_{ik} A(k \to r-1 \cap j \to s) & \text{if } r = m > s, \end{cases} \tag{5.5}$$

where m is the round in which teams i and j meet, and k runs over all 2^{r-1} possible round r opponents of team i.

The same method would compute $P(i \to r)$ and $P(i \to r \cap j \to s)$ if complete head-to-head data $\{p_{ij}\}$ was available. As noted earlier, usually most or all of the $\{p_{ij}\}$ are unknown.

Happily, $P(i \to r)$ is simply the fraction of opponents who have picked team i to win round r, and this information is available to pool organizers through simple counts. The organizers of large public NCAA men's basketball pools have a history of publishing the $P(i \to r)$ data. This is all that is needed for $\mu(X)$, $\text{cov}(X, Y)$, and $\text{cov}(X_\alpha, X_\beta)$ $(\alpha \neq \beta)$.

The final hurdle is the calculation of $\sigma^2(X)$, which requires $P(i \to r \cap j \to s)$ for all i, j, r, s. The probabilities $P(i \to r \cap j \to s)$ can be determined directly by examining poolsheets, where $P(i \to r \cap j \to s)$ is the proportion of opponents who chose teams i and j to reach and win rounds r and s, respectively. However, this information is less interesting to the public and unlikely to be published by pool organizers. An alternative ad hoc method would be to estimate p_{ij} from $P(i \to r)$ data and then compute $P(i \to r \cap j \to s)$ from p_{ij}. Finally, it may be that $\sigma^2(X)$ remains relatively unchanged over a range of reasonable inputs and could simply be taken as known. None of these methods is entirely satisfying, and the problem of $\sigma^2(X)$ remains the main difficulty in the practical computation of E^{norm}.

5.3. Finding Optimal Picks

Given $\{a_{ij}\}$, $\{p_{ij}\}$, and N, we want to find a bracket $\mathbf{b} \in \mathcal{O}$ that maximizes the approximate expected return $E^{\text{norm}}(\mathbf{b})$. A complete search of all possible picks is usually unreasonable because $|\mathcal{O}| = 2^{2^R - 1}$ for an R-round tournament.

Instead, we used a hill-climbing (greedy) algorithm based on the following definition of *neighbor picks*. Suppose that team i plays team j at some point in picks $\mathbf{y}$, with team i winning and eventually reaching round r. Let $\mathbf{y}'$ be identical to $\mathbf{y}$ except that team j reaches round r. Then, $\mathbf{y}$ and $\mathbf{y}'$ are neighbors. With this definition, every $\mathbf{y}$ has $2^R - 1$ neighbors, one for each game.

The principle of hill climbing is to begin with a set of picks and then calculate the expected return for each of its neighbor picks. Choose the best neighbor, and repeat the process until a local maximum is reached.

In experiments with NCAA tournament data, the hill-climbing process typically converged within 20–60 iterations. Although there is not always a unique local maximum, hundreds of random starting points consistently climb to the same few possibilities. A more sophisticated search seems unlikely to improve the situation, but some theoretical reason to trust hill climbing would be reassuring.

5.4. The NCAA Men's Basketball Tournament

We have tested our methods on the 2004 and 2005 NCAA Men's Basketball Tournaments. Our main sources of data were the large free online pools run by ESPN and by Yahoo. ESPN's Tournament Challenge received about five million entries in 2004, and Yahoo's Tournament Pick'em received about one million entries in 2005.

The 2004 tournament was already over when we began our analysis, and so we were able to automatically download 500,000 complete opponent brackets. Using this sample, we computed $P(i \to r)$ for every team i and round r by counting the number of opponents who actually chose team i to reach and win round r. Having a large supply of opponent poolsheets also allowed an accurate measure of $P(i \to r \cap j \to s)$ and therefore sidestepped the difficulties of computing $\sigma^2(X)$.

In 2005, we were able to generate picks in the three days between "Selection Sunday" and the tournament start on Thursday morning. Yahoo published the $P(i \to r)$ data, but we needed an ad hoc method to compute $P(i \to r \cap j \to s)$ and therefore $\sigma^2(X)$. We did this by estimating head-to-head pool probabilities for each pair of teams with

$$p_{ij} \approx \frac{1}{2} + \frac{1}{2}\left(\frac{P(i \to r)}{P(i \to r-1)} - \frac{P(j \to r)}{P(j \to r-1)}\right) \qquad (5.6)$$

for teams i and j, who meet in round r. Note that this gives the known correct value for teams that meet in Round 1. A complete report of the techniques, input data, and the various sets of picks generated for 2005 is available online in Clair and Letscher (2005).

In both years, we used three different sets of actual probabilities $\{a_{ij}\}$. Two were derived from computer rating systems (Massey 2004 and Sagarin 2004), where a_{ij} is computed as a function of the difference between the ratings of teams i and j. The third used historical results of matchups between teams with specific seedings.

Every choice of N, $\{a_{ij}\}$, $\{p_{ij}\}$, and scoring method w_r gives rise to a different expectation function E, and so the optimal picks vary with all of these inputs.

Figure 5 shows picks optimized for a 5,000,000 competitor pool with ESPN scoring and a_{ij} computed from Massey ratings. Within our model, the expected return on these picks is estimated at 798.8, and the correlation with

Figure 5. 2004 men's basketball picks ($N = 5{,}000{,}000$).

opponent scores is 0.15. In contrast, the expected return for the picks $\mathbf{b}_{\text{score}}$ that give the maximum expected score is only 32.7 because of a 0.37 correlation with opponent scores. Of 500,000 actual poolsheets from ESPN's pool, 4,297 had the same final four as $\mathbf{b}_{\text{score}}$, whereas only 49 final fours matched the picks in Figure 5.

As a crude measure of the shape of picks, one can count the number of favorites picked per round. Averaging these numbers for six data sets coming from two years and three possible $\{a_{ij}\}$ gives the favorites per round shown in Table 5. Even the small sample shows a clear trend.

5.5. The Opponent Model

To model opponents, we have made two key assumptions: first, that opponents pick using a Markov process; and second, that opponent scores are normally distributed. Simulation results presented in this section give evidence that the former is quite reasonable, but that the normality assumption has room for improvement.

First, we randomly selected 5,000 poolsheets entered (by humans) into ESPN's 2004 Tournament Challenge, and then simulated 10,000 tournaments. The frequency distribution of opponent scores is the solid black line in Figure 6, and has mean 678, standard deviation 190, and skewness 0.49. The normal distribution calculated from actual and pool probabilities has mean 673 and standard deviation 181. It is the dashed line in Figure 6. The gray line in Figure 6 shows scores for 5,000 poolsheets created by the Markov process and the pool probabilities.

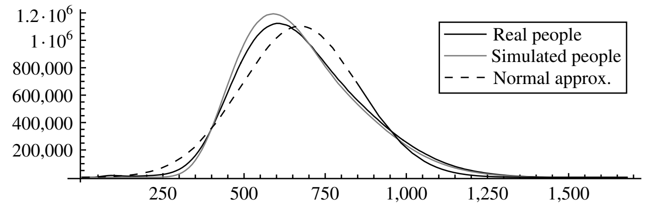
In a second experiment, for a fixed set of picks $\mathbf{y}$, we computed $E^{\text{norm}}(\mathbf{y})$ for $N = 100$ using our assumptions and pool probabilities from ESPN's 2004 Tournament Challenge. We then simulated 100,000 tournaments, randomly creating $n = 100$ opponent picks for each using the opponent model. Finally, we simulated 100,000 tournaments, randomly choosing $n = 100$ opponent picks made by human ESPN entrants for each. In both sets, the average return of $\mathbf{y}$ was recorded.

All of these simulations were repeated for 100 choices of $\mathbf{y}$, selected randomly from ESPN entrant picks. The calculated values $E^{\text{norm}}(\mathbf{y})$ were correlated with the simulated returns with an r-squared of 0.60 for simulated opponents and 0.53 for real opponents. Return against real opponents strongly correlates with return against simulated opponents, with an r-squared of 0.98.

Table 5. Favorites picked by various strategies.

Favorites	32	16	8	4	2	1
Optimal expected score	32	15.67	7.67	3.67	1.67	0.67
Optimal expected return, $N = 1{,}000$	31.83	15	7.33	2.33	0.5	0.5
Optimal expected return, $N = 5{,}000{,}000$	31	13.67	5.83	1.67	0.17	0.17

Figure 6. Opponent score distributions.



The independence assumptions and Markov model for opponent picks are quite reasonable because real and simulated opponents are nearly indistinguishable in the simulations above. On the other hand, the normal approximation fails to capture the left skewness of real pool scores, which comes from high point values for later-round games. It might be worth replacing the normal approximation with an appropriate skewed distribution, or even with a distribution calculated from simulated opponents.

5.6. Input Data

As with football picks, one would like some idea of how the optimal picks and their expected return are affected by variations in the inputs. Because there is only one tournament per year and there is so much input data, this a difficult question. In both 2004 and 2005, we used three different sources for $\{a_{ij}\}$. Because the bulk of the picks are still favorites, the different poolsheets are similar for early rounds. In 2004, the three poolsheets had different final fours, although all featured only teams seeded 1-3. All three also agreed that #1-seeded St. Joseph's was a good pick for the finals. In 2005, things were much more stable. We computed six sets of picks, using the three $\{a_{ij}\}$ and two choices of N and scoring system. All picked a Duke versus Washington final, and all agreed that heavy favorites Illinois and North Carolina should fail to reach the final four.

On the other hand, picks created with one set of $\{a_{ij}\}$ appear to be much poorer when evaluated using a different set. The expected return of 798.8 for the Massey-based picks in Figure 5 drops to 363.5 under Sagarin and 70.8 using NCAA historical seed records. Of course, the $\mathbf{b}_{\text{score}}$ picks also drop, from 32.7 to 8.3 and 0.4, respectively. As with football, it seems that the best picks remain good when inputs vary, whereas calculated values of expected return are less reliable.

6. Questions

How does one deal with different scoring methods? Our model assumes that all games in the same round are worth the same amount. However, some office pools give extra points for picking "upsets," or for special games (such as Monday night football). Notational complexity appears to be the only barrier to generalization, although providing incentives for upsets may produce a substantial change in opponent behavior. Some pools allow players to assign a confidence level to games, with scoring adjusted appropriately. This sort of pool is harder to understand because a new model of opponent behavior is needed.

What is the correct strategy for a pool with one opponent? This is interesting even if that opponent's picks are explicitly known ($p_{ij} \in \{0, 1\}\ \forall i, j$).

What if multiple entries are allowed? That is, which collection of picks $\mathbf{b}_1, \ldots, \mathbf{b}_k$ maximizes total winnings (given that a bet of k is now required)? Many pools allow multiple entries (ESPN in particular allows five), so this question is of practical interest. There is a straightforward upper bound for return on k picks of $kE(\mathbf{b})$ that might be approachable for small k, given the concentration of opponents in one small area of the picking space. Pursuing a multipick strategy might also lessen the dependence on the a_{ij}. To find a multiple-entry strategy, refinements to the normal approximation method are needed, although the main difficulty may simply be that the search space grows enormously.

Appendix

A.1. Computational Improvements

For picks $\mathbf{y}$ in a football pool, computing the exact formula (3.9) for $E(\mathbf{y})$ appears to require $O(4^g)$ terms. More precisely, there is a sum over 2^g outcomes $\mathbf{x}$, and for each outcome $\mathbf{x}$, $\mathscr{L}(\mathbf{x}, \mathbf{x} \wedge \mathbf{y})$ is a sum of about 2^g opponent bets. This section gives a method for computing $\mathscr{L}(\mathbf{x}, \mathbf{x} \wedge \mathbf{y})$ in polynomial time.

Proposition A.1. *For $s \geqslant 0$ and $\mathbf{x}$ the outcome where all favorites win,*

$$\mathscr{E}(\mathbf{x}, s) = \sum_{k=s}^{g} (-1)^{k-s} \binom{k}{s} \sigma_k(p), \tag{A1}$$

$$\mathscr{L}(\mathbf{x}, s) = 1 - \sum_{k=s}^{g} (-1)^{k-s} \frac{s}{k} \binom{k}{s} \sigma_k(p), \tag{A2}$$

where $\sigma_k(p)$ is the elementary symmetric polynomial in p_i of degree k.

Proof. We have $\mathscr{E}(\mathbf{x}, s) = \sum_{\mathbf{z}} P(\mathbf{z})$, where the sum is over all $\mathbf{z}$ with exactly s favorites and $g - s$ underdogs, and so has $\binom{g}{s}$ terms and is symmetric in p_i. Each term of the sum looks like

$$p_{\tau(1)} \cdots p_{\tau(s)} \cdot (1 - p_{\tau(s+1)}) \cdots (1 - p_{\tau(g)})$$

for some permutation τ, which multiplies to have $\binom{g-s}{k-s}$ terms in each degree $k \geqslant s$, each with sign $(-1)^{k-s}$. Because $\sigma_k(p)$ has $\binom{g}{k}$ terms, the degree k part of $\mathscr{E}(\mathbf{x}, s)$ must be $(-1)^{k-s}\binom{k}{s}\sigma_k(p)$.

The formula for $\mathscr{L}(\mathbf{x}, s)$ follows from

$$\mathscr{L}(\mathbf{x}, s) = \sum_{k=0}^{s-1} \mathscr{E}(\mathbf{x}, k)$$

and the identity

$$\sum_{i=0}^{j} (-1)^i \binom{n}{i} = (-1)^j \binom{n-1}{j}. \quad \square$$

Now we can compute $\mathscr{L}$ and $\mathscr{E}$ quickly (for numeric data). First, redefine the "favorite" so that $\mathbf{x}$ does pick all the favorites (replacing a_i and p_i with $1 - a_i$ and $1 - p_i$ as needed). Next, compute $S_k = \sum_i p_i^k$ for $k = 1, \ldots, g$. Finally, the Newton-Girard equations inductively compute $\sigma_k(p)$:

$$\sigma_k = (-1)^{k-1} \frac{1}{k} \sum_{i=0}^{k-1} (-1)^i \sigma_i S_{k-i}. \tag{A3}$$

The authors, sick of the four-hour wait for football picks, would love to find a way to eliminate the sum over all outcomes in (3.9).

A.2. Tournament Statistics

These results are generalizations of work in Kaplan and Garstka (2001), which computes the mean and variance for one fixed bet. All of the following formulas are readily computable, each with $O(T^2R^2)$ terms, where $T = 2^R$ is the number of teams.

Proposition A.2. *Suppose that one player makes one set of picks using probabilities $\{p_{ij}\}$, and has a score given by the random variable X. Then, the mean and variance of X are*

$$\mu(X) = \sum_{r=1}^{R} \sum_{i=1}^{T} w_r A(i \to r) P(i \to r), \tag{A4}$$

$$\sigma^2(X) = \sum_{r,s=1}^{R} \sum_{i,j=1}^{T} w_r w_s [A(i \to r \cap j \to s) P(i \to r \cap j \to s) - A(i \to r) A(j \to s) P(i \to r) P(j \to s)]. \tag{A5}$$

Proof. Write

$$X = \sum_g X_g,$$

where g runs over all $2^R - 1$ games, and X_g is a random variable giving the player's score in game g. The variables X_g usually have dependencies. In particular, let $\text{play}(g)$ be the set of teams that could play in a given game g. Then, X_g and X_h are independent when $\text{play}(g)$ and $\text{play}(h)$ are disjoint, and are otherwise dependent for generic a_{ij}, p_{ij}.

For g, h games in round r, s, respectively,

$$\mu(X_g) = w_r \sum_{i \in \text{play}(g)} A(i \to r) P(i \to r), \tag{A6}$$

$$\mu(X_h) = w_s \sum_{j \in \text{play}(h)} A(j \to s) P(j \to s), \tag{A7}$$

$$\mu(X_g X_h) = w_r w_s \sum_{i \in \text{play}(g)} \sum_{j \in \text{play}(h)} A(i \to r \cap j \to s) \cdot P(i \to r \cap j \to s). \tag{A8}$$

Because each team i can play in exactly one round r game, and each team j can play in exactly one round s game, we have

$$\mu\left(\sum_{\text{round}(g)=r} X_g\right) = w_r \sum_{i=1}^{T} A(i\to r)P(i\to r), \tag{A9}$$

$$\mu\left(\sum_{\substack{\text{round}(g)=r\\ \text{round}(h)=s}} X_g X_h\right) = w_r w_s \sum_{i,j=1}^{T} A(i\to r\cap j\to s) \cdot P(i\to r\cap j\to s). \tag{A10}$$

Summing over rounds r and s gives

$$\mu(X) = \mu\left(\sum_g X_g\right) = \sum_{r=1}^{R}\sum_{i=1}^{T} w_r A(i\to r)P(i\to r), \tag{A11}$$

$$\mu(X^2) = \mu\left(\sum_{g,h} X_g X_h\right) \tag{A12}$$

$$= \sum_{r,s=1}^{R}\sum_{i,j=1}^{T} w_r w_s A(i\to r\cap j\to s) \tag{A13}$$

$$\cdot P(i\to r\cap j\to s). \tag{A14}$$

Finally, $\sigma^2(X) = \mu(X^2) - \mu(X)^2$. □

We also need the covariance between the scores of two opponents and the covariance between our score and any one opponent. Both of these are specializations of the following:

PROPOSITION A.3. *Suppose that two independent pickers make picks with pool probabilities $\{p_{ij}\}$ and $\{q_{ij}\}$, and have scores given by the random variables X and Y. Then, the covariance*

$$\text{cov}(X,Y) = \sum_{r,s=1}^{R}\sum_{i,j=1}^{T} w_r w_s P(i\to r)Q(j\to s) \cdot [A(i\to r\cap j\to s) - A(i\to r)A(j\to s)]. \tag{A15}$$

PROOF. The calculation is nearly identical to the arguments for Proposition A.2. The only difference is that $P(i\to r\cap j\to s)$ is replaced by $P(i\to r)\cdot Q(j\to s)$ because the players are assumed to make picks independently. □

To get covariance between two opponents in a pool, take $Q = P$. To get covariance between an opponent and a fixed bet, take $Q(j\to s)\in\{0,1\}$ as appropriate.

Acknowledgments

The authors thank Dorit Hochbaum for her support and valuable suggestions and Tom Adams for his insightful correspondence. Also, thanks to David Moulton, Erin Langenstein, NetVision, and especially to ESPN for not shutting us down.

References

Adams, T. 2004. Poologic. www.poologic.com.

Boulier, B. L., H. O. Stekler. 1999. Are sports seedings good predictors? An evaluation. *Internat. J. Forecasting* **15**(1) 83–91.

Bradley, R. A., M. E. Terry. 1952. Rank analysis of incomplete block designs. I. The method of paired comparisions. *Biometrika* **39** 324–345.

Breiter, D. J., B. P. Carlin. 1997. How to play office pools if you must. *Chance* **10** 5–11.

Caudill, S. B. 2003. Predicting discrete outcomes with the maximum score estimator: The case of the NCAA men's basketball tournament. *Internat. J. Forecasting* **19** 313–317.

Caudill, S. B., N. H. Godwin. 2002. Heterogeneous skewness in binary choice models: Predicting outcomes in the men's NCAA basketball tournament. *J. Appl. Statist.* **29** 991–1001.

Clair, B., D. Letscher. 2005. The 2005 men's NCAA basketball tournament. euler.slu.edu/~clair/pools.

Dunnett, C. W., M. Sobel. 1955. Approximations to the probability integral and certain percentage points of a multivariate analogue of students t-distribution. *Biometrika* **42** 258–260.

Kaplan, E. H., S. J. Garstka. 2001. March madness and the office pool. *Management Sci.* **47**(3) 369–382.

Kaplan, E. H., M. J. Magazine. 2003. More March madness. *OR/MS Today* **30**(1) 38–42.

Kvam, P., J. Sokol. 2006. A logistic regession/Markov chain model for NCAA basketball. *Naval Res. Logist.* **53**(8) 788–803.

Massey, K. 2004. Massey ratings. www.mratings.com.

Metrick, A. 1996. March madness? Strategic behavior in NCAA basketball tournament betting pools. *J. Econom. Behav. Organ.* **30** 159–172.

Sagarin, J. 2004. Sagarin ratings. www.usatoday.com/sports/sagarin.htm.