Psychometric Modeling of Decision Making Via Game Play

Kenneth W. Regan
Department of CSE
University at Buffalo
Amherst, NY 14260 USA
regan@buffalo.edu

Tamal Biswas
Department of CSE
University at Buffalo
Amherst, NY 14260 USA
tamaltan@buffalo.edu

***Abstract*—We build a model for the kind of decision making involved in games of strategy such as chess, making it abstract enough to remove essentially all game-specific contingency, and compare it to known psychometric models of test taking, item response, and performance assessment. Decisions are modeled in terms of fallible agents Z faced with possible actions a_i whose utilities $u_i = u(a_i)$ are not fully apparent. The three main goals of the model are *prediction*, meaning to infer probabilities p_i for Z to choose a_i; *intrinsic rating*, meaning to assess the skill of a person's actual choices a_{i_t} over various test items t; and *simulation* of the distribution of choices by an agent with a specified skill set. We describe and train the model on large data from chess tournament games of different ranks of players, and exemplify its accuracy by applying it to give intrinsic ratings for world championship matches.**

Keywords. Computer games, chess, decision making, probabilistic inference, machine learning, statistics, fitting methods, maximum likelihood, psychometrics.

I. Introduction

In most applications of rational-choice theory to human decision making, the true or expected values and costs of actions are presumed to be known or knowable to the actors. The goal may be to find the set of choices that maximize (expected) utility values of benefits minus costs, or to resolve competing desires between actors. Alternately, the actors may express preferences that are taken as givens, and the goal is to determine a group preference. In *bounded rationality*, however, the true values are unknown, or may be too complex to obtain within a feasible timespan, and preferences may be based on fallacious perceived values.

We work in a bounded-rationality setting where "true values" are given by design, hindsight, or reasonable authority. This enables further goals of assessing an actor's past performance once the values come to light, and predicting future performance based on past record of finding optimal actions. This includes assessing the natural frequency of tangible mistakes even by the top performers—how often does "Homer nod"?

Test-taking is a prominent setting of this kind. The values of answer choices to questions are known by design. Often the values are binary (one choice correct, the rest wrong), but prescription of partial credits may lend an informative numerical value and ranking to every possible choice. The test taker—at least an honest one—does not know the values, and is graded on ability to find the best answers. Another is game-play, where either the results of moves are generally too complex to determine fully (as in chess) or may also depend on unknown chance allotments (as in bridge and poker). Where authoritative values of decision choices are unobtainable, owing to complexity or dependence on particulars of the opponents (such as bluffing in poker), the *results* of the actions—who wins the game or hand and by how much—furnish natural performance metrics.

The modeling in the paper marries the test-taking and game-playing settings. The elements are sets of decision events specified by available options and their authoritative values unknown to the actor, and internal parameters of the actor. Recent advances in computer chess playing programs enable one to obtain reliable values for the move options in chess positions on a large scale. Values given by the computer at different search depths, and other features in the data, may also enable measures of "difficulty" or "challenge" without requiring reference to chess notions such as attack, sacrifice, initiative, traps, and the like.

An old example of this marriage is a feature commonly called "Solitaire Chess" in newspapers or magazines[1]: The writer prescribes point values for various moves in the positions faced by the successful player in a recorded game, often but not always giving top value to the move that was played. The reader plays through the game (without peeking ahead), and in each indicated position selects the move he/she would play. Then the reader reveals the played move and points scoring and continues until the next position. At the end the reader's points total is scored on a scale determined by the preparer, for instance 150 points for "grandmaster," 120 for "master," 100 for "expert" and so on. This could be done with any set of chess positions used as a test, but having them come from one side in one game increases the popular element of "being in the game" and executing strategies across positions. The technical results of this and previous papers [1], [2] can be viewed as furnishing scientific scoring based on actual games played by grandmasters, masters, experts and so on, for evaluating moves made under real competition rather than "solitaire" or in simulations.

From the side of test-taking and related item-response theories [3], [4], [5], our work is an extension of Rasch modeling [6], [7], [8], [9] for polytomous items [10], [11], [12], [13], with similar mathematical ingredients (cf. [14], [15]). Rasch modeling has two main kinds of parameters, *person* and *item* parameters. These are often abstracted into the single parameters of actor *location* (or "ability") and item *difficulty*. The formulas in Rasch modeling enable predicting

[1] Bruce Pandolfini, famously played by Ben Kingsley in the movie "Searching for Bobby Fischer," writes this feature for the US *Chess Life* magazine.

978-1-4673-5311-3/13/$31.00 ©2013 IEEE

distributions of responses to items based on differences in these parameters.[2]

Our basic model has two person parameters called s for *sensitivity* and c for *consistency*. It employs a function $E(s, c)$, determined by regression from training data, mapping them to the standard *Elo rating* scale for chess players. The numbers on this scale have only relative significance but have arguably remained stable ever since its formal adoption in 1971 by the World Chess Federation (FIDE). The threshold for "master" is almost universally regarded as 2200 on this scale, with 2500 serving as a rating threshold for initial award of the title of Grandmaster, and 2000 called "Expert" in the Unites States. The current world champion, Viswanathan Anand, sports a rating of 2786, while top-ranked Magnus Carlsen's recent peak of 2872 is 21 points higher than the previous record of 2851 by former world champion Garry Kasparov. Computer programs running on consumer-level personal computers are, however, reliably estimated to reach into the 3200s, high enough that no human player has been backed to challenge a computer on even terms since then-world champion Vladimir Kramnik (currently 2803) lost a match in December 2006 to the Deep Fritz 10 program running on a quad-core PC. This fact has raised the ugly eventuality of human cheating with computers during games, but also furnishes the reliable values for available move options that constitute the only chess-dependent input to the model.

Our main departure from Rasch modeling is that these "hindsight utility values" are used to infer probabilities for each available response, without recourse to a measure of difficulty on the same rating scale. That is, we have no prior notion of "a position of Grandmaster-level difficulty," or "expert difficulty," or "beginning difficulty" *per-se*. Chess problems, such as White to checkmate in two moves or Black to move and win, are commonly rated for difficulty of solution, but the criteria are specialized. Instead we aim to *infer* difficulty from the expected loss of utility from that of the optimal move, and separately from other features of the computer-analysis data itself. Hence we propose a separate name for our paradigm: "Converting Utilities Into Probabilities."

The next sections attempt to formulate this paradigm abstractly, with minimal reference to chess. The main departure from skill rating in games [16], [17], [18], [19] is that the game *result* has no primary significance. A player may win a poorly played game through an opponent's blunder, or play heroically to save a draw which brings only a half-point reward. Shifting the scoring to individual move-decisions factors out the opponent's quality, and also multiplies the sample size by a large factor. A chess player may contest only 50 professional games in a given year, but these games may furnish 1,500 significant move decisions, even after discounting opening moves regarded as "book knowledge" and positions where one side has an overwhelming advantage. The upshot is to create a statistically robust formulation of "Intrinsic Performance Ratings" expressed on the standard Elo scale [1], [2].

[2]For a recent amusing blog post arguing identity between the Rasch and Elo chess rating formulas, see Christopher Long, "Baseball, Chess, Psychology and Psychometrics: Everyone Uses the Same Damn Rating System," 14 March 2013 entry on "The Angry Statistician" weblog, http://angrystatistician.blogspot.com/2013/03/baseball-chess-psychology-and.html

This paper aims to abstract this kind of decision-based assessment, identifying general features not dependent on chess. This will promote outside applications, such as intrinsic scoring of grades on tests with less reliance on either prior notions of difficulty or posterior "curving" after results are in. We do not achieve this here, but we demonstrate the richness and efficacy of our modeling paradigm. We show how it embraces both values and preference ranks, lends itself to multiple statistical fitting techniques that act as checks on each other, and gives consistent and intelligible results in the chess domain. Finally we discuss applying it toward more realistic simulation of fallible human game play.

II. Converting Utilities Into Probabilities

We are given a set T of decision events t, such as game turns in a chess game. Each event is specified by a list of some number ℓ of available options $a_1, \ldots, a_\ell$ known to the actor, and a list of values $U_t = (u_1, \ldots, u_\ell)$ of these actions that are not conveyed to the actor. The actor's skill at the game can informally be regarded as the degree to which he or she (or it—we can model fallible software agents the same way) perceives the authoritative values from situational information I_t (such as a chess position, or study notes for a test), and thereby selects the best option. That is to say, for any item t:

(a) Both the actor Z and the model are given a description of t and the options $a_1, \ldots, a_\ell$.

(b) The actor draws on experience and situational information I_t but does not know the values U_t.

(c) The model is given U_t, but does not use any application-specific information I_t.

(d) The model is either given values of the parameters $\vec{z} = z_1, z_2, \ldots$ defining proclivities of the actor Z, or has inferred such values from training data.

The model's *task* is to infer probabilities $P(\vec{z}) = (p_1, \ldots, p_\ell)$ for the actor Z to choose the respective options for the decision event t. We suppose that the labeling of moves is in nonincreasing order of the utilities given to the model, i.e., such that $u_1 \geq u_2 \geq \cdots \geq u_\ell$. We may also suppose that the number ℓ is the same for all turns t, by padding turns with $r < \ell$ stated options to have $\ell - r$ phantom options, each of "negatively infinite" value. Henceforth when we formally identify the actor Z with the probabilistic behavior mapping $P(\vec{z})$, we call this defining a *fallible computational agent*, along lines of [20], [21], [22], [23], [24].

In a simple rational-choice situation with perfect information about utility, and ignoring for now the possibility of two or more actions with equal best value, we would project $p_i = 1$ for the action a_i with highest u_i, and $p_j = 0$ for all other actions. In real-world situations of bounded rationality, however, we may expect to see substantial probability on a variety of reasonable options, and also on poor but deceptively attractive options. Can we model this so that over sufficiently many *turns* t at which an action must be chosen, and given parameters z quantifying the ability of Z to perceive the stipulated values u_i, we can make accurate projections of aggregate statistics? Here are three such statistics:

(a) *Best-Choice frequency* (BC). On a multiple-choice test, this is the score without partial credit, expressed as a percentage. In chess it is the frequency of move agreements with the computer; it is called MM in [1].

(b) *Aggregate Error* (AE). On an exam with partial credit, this is the total number of points lost. In chess it is the sum, over all moves (where the computer judges the player chose an inferior move), of the difference in value between the optimal move and the chosen one. Chess programs standardly give these values in units of *centipawns*—figuratively hundredths of a pawn, whereupon AE represents the total number of pawns "lost" by the players in the set of positions t. Where there is no confusion we also use AE for the per-move *average error*.

(c) *Ordinal Ranks* (OR). Besides the BC frequency f_1, these give the frequency f_2 of choosing the second-best option, f_3 for third-best, f_4 for fourth-best, and so on. Of course there may be no parity between second-best choices: some may be "close" while others are large errors. Projections of OR may take the difference in value into account, so that it is permissible to mix these kinds of data points. Likewise permissible is to assume all turns have the same number ℓ of options, padding those having fewer with "dummy options" having large negative values, which will translate to essentially-zero probability in projections.

Note that OR entails indexing choices by their values: $u_1 \geq u_2 \geq \cdots \geq u_\ell$. Also note that the projected probabilities $p_{t,i}$ over $t \in T$ alone suffice to generate projected values for these statistics, namely

$$\hat{bc} = \frac{1}{T}\sum_t p_{t,1} \quad (1)$$

$$\hat{ae} = \frac{1}{T}\sum_t p_{t,k}(u_1 - u_k) \quad (2)$$

$$\hat{f_k} = \frac{1}{T}\sum_t p_{t,k}. \quad (3)$$

To sum up, the paradigm is to have a function P of the personal parameters $\vec{z}$ and the utility values $u_{t,i}$. The personal parameters are assumed to be the same for all t in a given test-set T. The output $P(\vec{z}, \{ u_{t,i} \})$ is a matrix $(p_{t,i})$ giving for each decision event t the probabilities for choosing each of the options. A particular function P denotes an application-specific model within the paradigm.

Several reasonable axioms simplify and constrain the permissible functions P.

A. Axioms and Confidence Intervals

The first two axiomatic properties can be considered desirable, but the third has some questionable entailments.

(i) *Independence.* For all t, the generated values $(p_{t,1}, \ldots, p_{t,\ell})$ depend only on $(u_{t,1}, \ldots, u_{t,\ell})$ and $\vec{z}$.

(ii) *Extensionality of utility.* For all t and Z, $u_{t,i} = u_{t,j} \implies p_{t,i} = p_{t,j}$.

(iii) *Monotonicity.* For all t and i, if $u_{t,i}$ is replaced by a greater value u', then for all Z, $p'_{t,i} \geq p_{t,i}$.

Note that (iii) is different from saying that always $p_{t,1} \geq p_{t,2} \geq \cdots \geq p_{t,\ell}$, though that follows from (ii) and (iii) as-stated. The reason for doubting (iii) is the application to Z of all skill levels—it says that improving the hindsight quality of an answer makes it no less attractive, which runs counter to the idea in chess that "weaker players prefer weaker moves." One reason independence is desirable is that it yields inherent standard deviations and hence confidence intervals for projections of these statistics, in particular:

$$\sigma^2_{BC} = \sum_{t=1}^{T} p_{t,1}(1 - p_{t,1}) \quad (4)$$

$$\sigma^2_{AE} = \frac{1}{T}\sum_{t=1}^{T}\sum_{i\geq 2} p_{t,i}(1 - p_{t,i})(u_{t,1} - u_{t,i}). \quad (5)$$

One way of attempting to cope with having less than full independence is to regard the effective sample size $|T|$ as having shrunk, and hence to widen the inferred error bars by a corresponding factor. Empirical runs in the chess application [2] have suggested a 1.15 multiplier for σ_{BC} and a 1.4 multiplier for σ_{AE}. These factors also represent allowance for modeling error.

III. Training Data and Fitting Metrics

Every piece of training data is an *item*

$$I = (u_1, \ldots, u_\ell; i; e)$$

where i is the index of the option that was chosen, and e gives supplementary information about the person or agent who made the choice. In examinations, e may be prior information about past grades or test results, or alternatively may be posterior information such as the overall score on the exam itself. In chess, e can be the Elo rating of the player making the move. We index an item and its components by the turn t, and sometimes write just t in place of I, calling the item a *tuple*.

In this paper we postulate a mapping $E(Z)$ from the agent space to the values e that come with the training data. This mapping need not be invertible—indeed when two or more scalar parameters comprise $\vec{z}$ this is not expected. Its main point is that when regressing on a subset of items whose values e are all equal (or close to equal) to obtain z, comparing $E(z)$ and e acts as a check on the results. They need not be equal—perhaps E itself has been fitted by a final linear regression against e and the particular e is an outlier in this fit—but lack of closeness is a reason to doubt the method used to fit z.

When the estimation method does not guarantee that the projected means agree with the sample means, for BC and AE in particular, then the difference from the sample becomes a measure of goodness (or rather badness) of fit. We express the deviations from the sample means as multiples of the inherently projected standard deviations, that is as multiples of σ_{BC} and σ_{AE}. Technically this assumes independence of turns, but this assumption is not a strong consideration because

we are not using them to infer z-scores for hypothesis tests. They are mainly a convenient choice of units when comparing results between training sets of different sizes.

Our main independent metric for goodness of fit involves the projected ordinal frequencies $\hat{f}_k$ for $1 \leq k \leq \ell$ and the sample values f_k. The *ordinal rank fit* (ORF) is given by

$$\sum_{k=1}^{\ell} (\hat{f}_k - f_k)^2.$$

In tables, the frequencies are expressed as percentages—equivalently, the ORF score is multiplied by 10,000. We do not weight ORF by the number of items with $i = k$ (i.e., by f_k itself), but we consider a fitting method that tries to minimize this score.

The metrics and training and fitting methods all extend naturally when items I_t for different turns t are given different weights w_t. The weight w_t may represent some measure of the value or difficulty of the decision. Our experiments in the basic model reported here use unit weights.

IV. Fitting Methods

In all cases we are given training data consisting of items. Since the data are fixed, we can regard the probabilities $p_{t,j}$ as functions of Z alone.

A. MLE and Bayes

Maximum Likelihood Estimation fits $\vec{z}$ to maximize the probability of the selected options i_t in the training data. By independence this means to maximize

$$\prod_t p_{t,i_t},$$

which is equivalent to minimizing the log-sum

$$\sum_t \ln(1/p_{t,i_t}).$$

We write z_{ML} for some value of Z that minimizes this log-sum, and call $P(z_{ML})$ the *max-likelihood probabilities*. Bayesian methods likewise apply, and were introduced for chess by Haworth [25], [26].

For completeness, we derive the result that Bayesian iteration approaches the ML estimator in this setting. Let $A(z)$ denote the event that the agent in the training data with chosen options $\vec{i} = i_1, i_2, \ldots$ arises from $Z = z$. By Bayes' Theorem, assuming the space Z is finite,

$$\begin{aligned}
\Pr(A(z) \mid \vec{i}) &= \frac{\Pr(\vec{i} \mid A(z))\Pr(A(z))}{\Pr(\vec{i})} \\
&= \frac{\Pr(\vec{i} \mid A(z))\Pr(A(z))}{\sum_z \Pr(A(z))\Pr(\vec{i} \mid A(z))} \\
&= \frac{\prod_t \Pr(i_t \mid A(z))\Pr(A(z))}{\sum_z \Pr(A(z)) \prod_t \Pr(i_t \mid A(z))} \\
&= \frac{\prod_t p_{t,i_t}(z)\Pr(A(z))}{\sum_z \prod_t p_{t,i_t}(z)\Pr(A(z))}.
\end{aligned}$$

The standard "know-nothing prior" assumption $\Pr(A(z)) = 1/|Z|$ lets us simplify this even further to

$$\Pr(A(z) \mid \vec{i}) = \frac{\prod_t p_{t,i_t}(z)}{\sum_z \prod_t p_{t,i_t}(z)}.$$

Note that the global independence assumption not only creates a simple product over t but also makes the value independent of the order of presentation of the data for each t. Thus the Bayesian probability of $Z = z$ is just the normalized likelihood function.

Write N_z for $\prod_t p_{t,i_t}(z)$. Upon iterating the data d times, we get

$$\Pr(A(z) \mid \vec{i}^d) = \frac{N_z^d}{\sum_z N_z^d}.$$

Because $a^d = o(b^d)$ whenever $a < b$ and the vector of values N_z is finite, the right-hand side as $d \to \infty$ converges pointwise to the Dirac delta function for the value z maximizing N_z, which is just $z = z_{ML}$ as before. (This also holds true under any fixed prior $A(z)$.)

Thus the peaks of the Bayesian probability curves approach the ML estimators. Large homogeneous training sets can be expected to behave like d-fold iterations of a smaller training set. Note that in both cases, only the probabilities of the selected options a_{t,i_t} are involved in the formulas. The ordinal information in i_t is not used. At least in this formulation, it seems that the MLE and Bayesian approaches are not using all available information.

We move on to simple frequentist approaches. We define $d(x, y)$ to be (the square of) a distance function, not necessarily supposing $d(x, y) = (x - y)^2$ for use with least-squares estimation. Since there is no notion of "same outcome," the issue becomes how best to preserve the intuition of building frequencies for the outcomes. One idea is to impose a percentile grid on them.

B. Percentile Fitting

The "Percentile Fitting" method of [1] attempts to avoid these choices and weighting issues. The method is to minimize a distance integral of the form

$$\int_{q=0}^{q=1} d(q, f_q(z))$$

where $f_q(z)$ is the *hit score for percentile* q defined as follows. The hit score is the average of the hit scores for each tuple t, so suppressing t we need only define $f_q(z)$ for one vector of projected probabilities $P(z) = (p_1(z), p_2(z), \ldots, p_\ell(z))$. Here is where the fixed ordering of outcomes is used. Let $i = i_t$ be the selected outcome for that tuple. Define

$$\begin{aligned}
p &= \sum_{j=1}^{i-1} p_j(z); \qquad r = p + p_i(z), \\
f_q(z) &= \begin{cases} 1 & \text{if } q \geq r \\ \frac{q-p}{r-p} & \text{if } p \leq q \leq r \\ 0 & \text{if } q \leq p. \end{cases}
\end{aligned}$$

Here is the frequentist intuition. Consider any fixed value of q, say $q = 0.60$, and consider any projected tuple

$(p_1(z), p_2(z), \ldots, p_\ell(z))$. The parameter(s) z represent a way of stretching or compressing sub-intervals of width $p_k(z)$ in the unit interval. Let us suppose first that q is exactly at the upper end of interval p_k, meaning that $p_1(z)+p_2(z)+\cdots+p_k(z) = q$. Then we interpret z as representing the assertion that the probability of one of the first k options being chosen is exactly q. That is to say, if $i \leq k$ then we call the tuple a "hit," else it is a "miss." So this particular z is asserting that the probability of a hit is q, and that is the z that we wish to find.

If q sits midway inside interval p_k, then we must consider how to score z in the case $i = k$. To interpolate correctly, let b be the ratio of the real-number distance of q from the left end of the interval to its width p_k. Then score this case as b of a hit. Thus z and q represent the assertion that the expected hit score for the tuple *at* percentile q is none other than q itself.

For each z and q, this prescription defines a criterion for scoring a hit for each tuple, and asserts that this expectation is q. Since the expectation is the same for each tuple, we have intuitively achieved the effect of the simple-frequency case, and can aggregate over the tuples. The frequency function $f_q(z)$ defined above tabulates the actual hit scores from the data. The degree of fit given by z for percentile q is then quantified as the distance between q itself and $f_q(z)$.

Treating q itself as a continuous parameter leads to minimizing the above integral. The one arbitrary choice we see is whether this should be weighted in terms of q. Minimizing

$$\int_{q=0}^{q=1} H(q)d(q, f_q(z))$$

instead is natural because having $H(0) = H(1) = 0$ reinforces the idea that the hit percentage projections are automatically correct at the endpoints $q = 0$ and $q = 1$. Apart from this, our intuitive point of using percentiles is that they skirt issues of skedasticity. We abbreviate this method as PF.

C. Fitting to Equate Means for BC and AE

The projected probabilities also yield a *projected utility* $u(z) = \sum_j u_j p_j(z)$. This can readily be summed or averaged over all tuples. Thus one can also fit z by equating the projected $u(z)$ with the actual utility u achieved in the training data. This is affinely related to minimizing the AE metric.

In cases where the objective is to see how often the agent makes the optimal choice, as well as modeling its average (falloff in) utility (from optimal), one can write two equations in the parameters Z. When Z comprises just two parameters, one can fit by solving two equations in two unknowns, equating $u = u(z)$ and the first-choice hit frequency $h_1 = |\{\, t : i_t = 1 \,\}|/T$ with the average of $p_{t,1}(z)$. This hybrid fitting method bypasses all of the above options, and hence acts as a helpful check on them. We abbreviate this FF for "first-choice and falloff." FF is the method chosen for computing IPRs below.

D. Fitting the Ordinal Ranks

We can fit to minimize the ORF statistic, or alternatively its frequency-weighted version $\sum_k f_k(f_k - \hat{f}_k)^2$. For data such as ours where about half the "mass" is on index 1—that is, when the best answer or trade or chess move is found at least half the time (at least when items have unit weights)—the latter policy is a compromise on simply solving to make the BC projection agree with the sample mean. The compromise policy avoids over-fitting, and avoids heteroskedasticity issues with ORF itself. Minimizing ORF emphasizes the importance of getting accurate projections for the "tail" of the distribution of choices: bad mistakes on exams or "blunders" at chess.

V. Some Theoretical Considerations

In case the parameter space for Z allows a dense set of probability vectors, the simple case of repeated data (or equal utility vectors) allows exact fitting, and gives the same optimal z under any method.

Theorem 5.1: Percentile Fitting agrees with Maximum Likelihood for homogeneous data and free parameterization of probabilities.

Proof: Take $f_1, \ldots, f_\ell$ to be the frequencies from the data. The log-sum minimand for MLE then becomes

$$e(z) = f_1 \ln(1/p_1(z)) + f_2 \ln(1/p_2(z)) + \cdots f_\ell \ln(1/p_\ell(z)).$$

This is known to be minimized by setting $p_j(z) = f_j$ for each j, in accordance with the basic frequency idea, and the freedom assumption on z allows this to be realized. It remains to show that PF achieves the same minimum z.

For this z, let q be any percentile. If q falls on the endpoint r of any interval $p_k = p_k(z)$, then as $r = p_1 + p_2 + \cdots p_k = f_1 + f_2 + \cdots f_k$, the training data gives $f_q(z) = r = q$. Since other values of q occupy the same medial position in the same interval over all of the equal tuples, the interpolation gives $f_q(z) = q$ for all q, so the PF minimand is zero.

Also $h_1 = f_1 = p_1(z)$ and $u = \sum_j u_j f_j = \sum_j u_j p_j(z) = u(z)$, so the FF equations hold.

■

A. Differences Among MLE/Bayes and PF

Now we give an example showing that this equivalence can be disturbed by constraining the parameters. Indeed it seems to be the simplest example that can possibly show a difference. Let each tuple have outcomes m_1, m_2, m_3, and let the probabilities be given by $p_1(z) = p_2(z) = z$, $p_3(z) = 1 - 2z$, with one numerical parameter $z \in [0, 1]$. Consider training data with $t = 2$ equal tuples, in which m_1 and m_3 are chosen once each. The ML maximand is $z(1 - 2z)$ and is maximized at $z = 1/4$.

Percentile fitting gives a different answer, however. The PF minimand is a three-piece integral. The first piece integrates $d(q, \frac{1}{2}\frac{q}{z})$ from $q = 0$ to $q = z$. The second piece integrates $d(q, \frac{1}{2})$ from $q = z$ to $q = 2z$. The third piece integrates $d(q, \frac{1}{2} + \frac{1}{2}\frac{q-2z}{1-2z})$ from $q = 2z$ to $q = 1$. For $d(x, y) = (x - y)^2$, symbolic computation with *Mathematica™* shows this is minimized for $z = 3/10$, not $z = 1/4$.

VI. Application Model and Experimental Domain

Chess has been a popular experimental domain for many studies aspiring to have more general significance. Here are several reasons pertaining in particular to the prior work [27], [28], [1]:

1) Chess games and results are public—there are no copyright or privacy considerations.
2) The decisions are taken under conditions of actual competition, not simulations.
3) The human subjects have similar training and backgrounds, and the games have no systematic or substantial outside influences (put another way, the modeling can be free of "nuisance terms").
4) There is a well-defined skill metric, namely the chess Elo rating system.
5) The utility values are assigned by a recognized human-neutral authority, namely the champion chess program Rybka 3 [29].
6) The data sets are unique—comprising *all* games recorded between players with ratings e near the same Elo century mark in official chess events under "standard individual tournament conditions" in a specified time period. There is no arbitrariness of choosing data from certain kinds of exams or kinds of financial markets.
7) The data sets and statistical analyzing code are freely available by request, though they are not (yet) public.
8) The training sets each contain over 5,000 data points, some over 20,000.

To reprise some details from [1], [2], the defining equation of the particular model used there and here is the following, which relates the probability p_i of the i-th alternative move to p_0 for the best move and its difference in value:

$$\frac{\log(1/p_i)}{\log(1/p_0)} = e^{-\left(\frac{\delta}{s}\right)^c}, \quad \text{where} \quad \delta_i = \int_{v_i}^{v_0} \frac{1}{1+|z|}dz. \quad (6)$$

Here when the value v_0 of the best move and v_i of the i-th move have the same sign, the integral giving the scaled difference simplifies to $|\log(1+v_0) - \log(1+v_i)|$. This employs the empirically-determined logarithmic scaling law.

The skill parameters are called s for "sensitivity" and c for "consistency" because s when small can enlarge small differences in value, while c when large sharply cuts down the probability of poor moves. The equation solved directly for p_i becomes

$$p_i = p_0^{\alpha} \qquad \text{where} \qquad \alpha = e^{-\left(\frac{\delta}{s}\right)^c}. \quad (7)$$

The constraint $\sum_i p_i = 1$ thus determines all values. By fitting these derived probabilities to actual frequencies of move choice in training data, we can find values of s and c corresponding to the training set.

Once we have s and c, these equations give us *projected probabilities* $p_{i,t}$ for every legal move m_i in the position at every relevant game turn t. Per arbitrary choice we *omit*: game turns 1–8, turns involved in repetitions of the position, and turns where the program judges an advantage greater than 300 centipawns for either side. These and some further specific modeling decisions are given detail and justification in [1].

VII. Computing Intrinsic Performance Ratings

The training data comprised all recorded games from the years 2006–2009 in standard individual-play tournaments, in which both players had current Elo ratings within 10 points of a century mark, 2200 through 2700. Table I gives the values of AE_e (expected AE) that were obtained by first fitting the training data for 2006–09, to obtain s, c, then computing the expectation for the union of the training sets. It was found that a smaller set S of moves comprising the games of the world championship tournaments and matches from 2005 to 2008 gave identical results to the fourth decimal place, so S was used as the fixed "Solitaire Set." An improved way of treating potential repetitions compared to [1], [2] produces the following table:

Elo	2700	2600	2500	2400	2300	2200
AE_e	.0561	.0637	.0707	.0744	.0860	.0917

TABLE I. Elo-AE correspondence

A simple linear fit then yields the rule to produce the Elo rating for any (s, c), which we call an "Intrinsic Performance Rating" (IPR) when the (s, c) are obtained by analyzing the games of a particular event and player(s).

$$\text{IPR} = 3475 - 13896 \cdot AE_e. \quad (8)$$

This expresses, incidentally, that at least from the vantage of Rybka 3 run to reported depth 13, perfect play has a rating under 3500, which is only 600 points above Carlsen. This is reasonable considering that if a Carlsen can draw even once in twenty-five games against perfect play, or a 2800 player like Kramnik once in fifty, the Elo rating formula can never give it a higher rating than that.

The procedure for computing an Intrinsic Performance Rating (IPR) given a set of games by a particular player in a chess event is hence the following:

1) Analyze the player's moves in these games with one or more strong chess programs such as Rybka.
2) Run the parameter regression on these moves to fund the best-fitting values of the player parameters s and c.
3) Use the inferred (s, c) to generate the expected average-error for $Z_{s,c}$ on the fixed "Solitaire Set" S.
4) Plug the AE_e value from step 3 into the equation (8) to obtain the IPR.
5) Also estimate the relative width of the error bars for AE_e on S to be the same as the relative error in the AE statistic projected for the player's own moves in the given games.
6) Use the same conversion (8) to place the error bars on the same Elo scale.

The error-analysis methods of Guid and Bratko [30], [31] neglect the importance of normalizing the AE statistic to a common reference set S. They compute AE separately for different players on their respective sets of positions. When they are facing each other in a match this may be reasonable, but when they are playing different opponents, one game may have a significantly higher expectation of error owing to less-placid nature of its positions. The above procedure handles this issue by computing separate regressions for (s, c) and (s', c') for the respective players that take the nature of their respective positions into account, and then compare the projections for the corresponding agents $Z_{s,c}$ and $Z_{s',c'}$ on the same positions in S. Nor do their methods provide error bars or a trained translation to the Elo rating scale at all; they themselves only justify significance of the ranking order they obtain, rather than an absolute quality scale such as Elo [32].

To illustrate the IPR method, here is a table of IPRs for all world chess championship matches since 1972.

Match	Elo	IPR	$2\sigma_a$ range	diff	#moves
Wch 1972	2723	2682	2560–2803	-41	1367
Fischer	2785	2751	2584–2919	-34	680
Spassky	2660	2613	2439–2788	-47	687
Wch 1974	2685	2660	2553–2767	-25	1787
Karpov	2700	2692	2541–2842	-8	889
Korchnoi	2670	2630	2478–2782	-40	898
Wch 1978	2695	2686	2601–2772	-9	2278
Karpov	2725	2668	2551–2785	-57	1141
Korchnoi	2665	2703	2580–2826	+38	1137
Wch 1981	2698	2738	2622–2854	+40	1176
Karpov	2700	2835	2680–2990	+135	587
Korchnoi	2695	2639	2470–2808	-56	589
Wch 1984	2710	2816	2744–2888	+106	2446
Karpov	2705	2809	2707–2912	+104	1219
Kasparov	2715	2823	2722–2924	+108	1227
Wch 1985	2710	2696	2586–2806	-14	1420
Karpov	2720	2660	2505–2816	-60	712
Kasparov	2700	2733	2579–2888	+33	708
Wch 1986	2723	2788	2683–2893	+65	1343
Karpov	2705	2747	2604–2891	+42	670
Kasparov	2740	2828	2676–2981	+88	673
Wch 1987	2720	2659	2547–2771	-61	1490
Karpov	2700	2731	2577–2885	+31	742
Kasparov	2740	2577	2416–2738	-163	748
Wch 1990	2765	2669	2557–2782	-96	1622
Karpov	2730	2647	2477–2817	-83	812
Kasparov	2800	2693	2546–2841	-107	810
Wch 1993	2730	2628	2492–2765	-102	1187
Kasparov	2805	2639	2450–2828	-166	593
Short	2655	2618	2421–2815	-37	594
Wch 1995	2760	2726	2548–2904	-34	767
Anand	2725	2612	2337–2886	-113	382
Kasparov	2795	2832	2611–3054	+37	385
Wch 2000	2810	2833	2706–2960	+23	867
Kasparov	2849	2786	2589–2984	-63	435
Kramnik	2770	2883	2724–3043	+113	432
Wch 2004	2756	2844	2712–2976	+88	726
Kramnik	2770	2857	2681–3033	+87	363
Leko	2741	2831	2631–3031	+90	363
Wch 2006	2778	2744	2629–2860	-34	896
Kramnik	2743	2727	2575–2879	-16	445
Topalov	2813	2754	2584–2925	-57	451
Wch 2008	2778	2709	2526–2892	-69	545
Anand	2783	2867	2645–3089	+84	271
Kramnik	2772	2534	2246–2821	-238	274
Wch 2010	2796	2735	2606–2865	-61	985
Anand	2787	2766	2574–2959	-21	491
Topalov	2805	2703	2530–2876	-102	494
Wch 2012	2759	2949	2801–3097	+190	495
Anand	2791	2987	2795–3180	+196	249
Gelfand	2727	2907	2678–3136	+180	246
Averages	2741	2739		-2	1,261
Move-wtd.	2730	2726		-4	
Omni run	2730	2728	2699–2756	-2	21,397

The IPR figures averaged over all the matches come strikingly close to the average of the players' ratings. This remains true when the average is weighted by the number of moves in the respective matches. The near-equality is also witnessed when all the moves of all the matches are thrown together into one large set, on which a single regression is performed.

What is significant there is that the world championship matches are disjoint from the training sets, which comprise games from tournaments only—round-robin or small Swiss-system events. These games ranged from the Elo 2200 to the Elo 2700 levels, by and large below the standards of all the matches. Thus the model is giving accurate performance assessment even under extrapolation to higher skill levels. The observed closeness is also much finer than the computed width of the error bars for the single large run of all the moves.

VIII. Simulating Human Game Play

Many chess programs implement playing at Elo levels E below their full strength by limiting their search time or depth. As originally conceived by Haworth [25], [26], we propose instead to simulate $P(s, c)$ for parameter values s, c corresponding to E. Use enough time to obtain reliable move values, then use (s, c) to generate the model's move probabilities $(p_1, \ldots, p_\ell)$, and select a move at random from this distribution. If the FF fitting method was used to obtain s and c, the distribution will give the same first-move agreement and average error.

The resulting fallible computational agent $P(s, c)$ will thus play inferior moves with similar frequency to the actual human players in the training sets giving these parameters. It will often make moves reflecting a longer search horizon than the time/depth-limited simulation, but will also make some crass errors. How well this resembles actual human experience—the 2400-rated first author can vouch for his own frequency of simple oversights—is food for further behavioral study. One remaining lack compared to human play is that the full independence of these random move selections may give a "plan-less" feel. If so, it is possible but challenging to expand the model to consider two-move or three-move sequences as units.

IX. Conclusions and Future Extensions

We hope to spur from this a deeper comparative examination of methods used in psychometric test scoring, and other application areas such as financial analysis. We also speculate that the skill assessment methods used in chess can be carried over to these other domains, even without the elements of game-play and win-lose-draw results. They apply to direct evaluation of decision quality instead, but inherit the interpretive power of the chess Elo rating system. The rigor and definiteness of the chess model and its data also strengthen confidence in the mathematical underpinnings of this kind of modeling.

We infer the difficulty of a game turn via the projected AE statistic for the turn. One drawback is that this projection depends on the parameters of the player. In one sense this is unavoidable, since usually weaker players are projected to make more errors. The difficulty measure comparing two turns t_1 and t_2 would then look at the global differences in the mappings g_1, g_2 from (s, c) to the respective AE expectations.

It would be better to find a measure of difficulty that is intrinsic to the decision event and its utility vector, and obtain a hoped-for scientific result that this implies observed differences in the statistic for various (s, c) locations. One approach we are trying extends the model to incorporate the notion of the *depth* or *time* needed to reach a determination, in a way that also applies to the utility values themselves, not just the actor's effort to perceive them. Chess programs conduct their searches by iterating to progressively higher depths d. Often a possible move will "swing" in value considerably up or down as d increases. This corresponds in test-taking to the idea of a "tricky" question (not necessarily a "trick question") that is designed to make a poor answer seem plausible until the test-taker reflects further on the question.

To reflect this on the player side, we can augment the model by creating a third personal parameter d representing the player's most typical peak depth of thinking. That there is a correspondence between program-depth values and measurable depth of thinking (in substantially time-limited play) is established by Moxley et al. [33]. Hence we expect to improve the player modeling by using the evaluation of all legal moves at each depth by the chess program. The larger anticipated payoff is that the total amount of "swing" among moves in a given position—simply the variance in values with respect to depth—may furnish an intrinsic measure of difficulty.

The notion of depth/time should also help determine how people perform in time-constrained environments. For examples in multiple-choice questions with partial credit, one may expect the decision made in time constrained environment would be worse than the unhurried ability of the responder. An examinee in a time-constrained environment may need to trust intuition more than deliberate thinking. We have analyzed games from fast-chess events to show dropoffs in IPR of several hundred Elo points, but it would be even better to be able to predict the dropoff with time curve as a function of the d parameter. This would shed finer light on issues noted by Chabris and Hearst [34].

REFERENCES

[1] K. Regan and G. Haworth, "Intrinsic chess ratings," in *Proceedings of AAAI 2011, San Francisco*, 2011.

[2] K. Regan, B. Macieja, and G. Haworth, "Understanding distributions of chess performances," in *Proceedings of the 13th ICGA Conference on Advances in Computer Games*, 2011, tilburg, Netherlands.

[3] F. B. Baker, *The Basics of Item Response Theory*. ERIC Clearinghouse on Assessment and Evaluation, 2001.

[4] G. L. Thorpe and A. Favia, "Data analysis using item response theory methodology: An introduction to selected programs and applications," *Psychology Faculty Scholarship*, p. 20, 2012.

[5] G. A. Morris, L. Branum-Martin, N. Harshman, S. D. Baker, E. Mazur, S. Dutta, T. Mzoughi, , and V. McCauley, "Testing the test: Item response curves and test quality," *American Journal of Physics*, vol. 81, no. 144, 2013.

[6] G. Rasch, *Probabilistic models for for some intelligence and attainment tests*. Copenhagen: Danish Institute for Educational Research, 1960.

[7] ——, "On general laws and the meaning of measurement in psychology," in *Proceedings, Fourth Berkeley Symposium on Mathematical Statistics and Probability*. University of California Press, 1961, pp. 321–334.

[8] E. Andersen, "Conditional inference for multiple-choice questionnaires," *Brit. J. Math. Stat. Psych.*, vol. 26, pp. 31–44, 1973.

[9] D. Andrich, *Rasch Models for Measurement*. Beverly Hills, California: Sage Publications, 1988.

[10] ——, "A rating scale formulation for ordered response categories," *Psychometrika*, vol. 43, pp. 561–573, 1978.

[11] G. Masters, "A Rasch model for partial credit scoring," *Psychometrika*, vol. 47, pp. 149–174, 1982.

[12] J. M. Linacre, "Rasch analysis of rank-ordered data," *JOURNAL OF APPLIED MEASUREMENT*, vol. 7, no. 1, 2006.

[13] R. Ostini and M. Nering, *Polytomous Item Response Theory Models*. Thousand Oaks, California: Sage Publications, 2006.

[14] F. Wichmann and N. J. Hill, "The psychometric function: I. Fitting, sampling, and goodness of fit," *Perception and Psychophysics*, vol. 63, pp. 1293–1313, 2001.

[15] H. L. J. V. D. Maas and E.-J. Wagenmakers, "A psychometric analysis of chess expertise," *American Journal of Psychology*, vol. 118, pp. 29–60, 2005.

[16] A. Elo, *The Rating of Chessplayers, Past and Present*. New York: Arco Pub., 1978.

[17] M. E. Glickman, "Parameter estimation in large dynamic paired comparison experiments," *Applied Statistics*, vol. 48, pp. 377–394, 1999.

[18] P. Dangauthier, R. Herbrich, T. Minka, and T. Graepel, "TrueSkill through time: Revisiting the history of chess," Microsoft Report 74417, research.microsoft.com/pubs/74417/NIPS2007_0931.pdf, 2007, poster, 2007 Neural Information Processing (NIPS) workshop.

[19] T. I. Fenner, M. Levene, and G. Loizou, "A discrete evolutionary model for chess players' ratings," *IEEE Trans. Comput. Intellig. and AI in Games*, vol. 4, no. 2, pp. 84–93, 2012.

[20] A. Reibman and B. Ballard, "Non-minimax strategies for use against fallible opponents," in *proceedings, Third National Conference on Artificial Intelligence (AAAI-83)*, 1983.

[21] R. Korf, "Real-time single-agent search: first results," in *Proceedings, 6th International Joint Conf. on Artificial Intelligence*, 1987.

[22] ——, "Real-time single-agent search: new results," in *Proceedings, 7th International Joint Conf. on Artificial Intelligence*, 1988.

[23] ——, "Generalized game-trees," in *Proceedings, 8th International Joint Conf. on Artificial Intelligence*, 1989.

[24] P. Jansen, "KQKR: Awareness of a fallible opponent," *ICCA Journal*, vol. 15, pp. 111–131, 1992.

[25] G. Haworth, "Reference fallible endgame play," *ICGA Journal*, vol. 26, pp. 81–91, 2003.

[26] ——, "Gentlemen, Stop Your Engines!" *ICGA Journal*, vol. 30, pp. 150–156, 2007.

[27] G. DiFatta, G. Haworth, and K. Regan, "Skill rating by Bayesian inference," in *Proceedings, 2009 IEEE Symposium on Computational Intelligence and Data Mining (CIDM'09), Nashville, TN, March 30–April 2, 2009*, 2009, pp. 89–94.

[28] G. Haworth, K. Regan, and G. DiFatta, "Performance and prediction: Bayesian modelling of fallible choice in chess," in *Proceedings, 12th ICGA Conference on Advances in Computer Games, Pamplona, Spain, May 11–13, 2009*, ser. Lecture Notes in Computer Science, vol. 6048. Springer-Verlag, 2010, pp. 99–110.

[29] V. Rajlich and L. Kaufman, "Rybka 3 chess engine," 2008, http://www.rybkachess.com.

[30] M. Guid and I. Bratko, "Computer analysis of world chess champions," *ICGA Journal*, vol. 29, no. 2, pp. 65–73, 2006.

[31] ——, "Using heuristic-search based engines for estimating human skill at chess," *ICGA Journal*, vol. 34, no. 2, pp. 71–81, 2011.

[32] M. Guid, A. Pérez, and I. Bratko, "How trustworthy is Crafty's analysis of world chess champions?" *ICGA Journal*, vol. 31, no. 3, pp. 131–144, 2008.

[33] J. H. Moxley, K. A. Ericsson, N. Charness, and R. T. Krampe, "The role of intuition and deliberative thinking in experts' superior tactical decision-making," *Cognition*, vol. 124, no. 1, pp. 72 – 78, 2012. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0010027712000558

[34] C. Chabris and E. Hearst, "Visualization, pattern recognition, and forward search: Effects of playing speed and sight of the position on grandmaster chess errors," *Cognitive Science*, vol. 27, pp. 637–648, 2003.