LAURA MARTIGNON and ULRICH HOFFRAGE

FAST, FRUGAL, AND FIT: SIMPLE HEURISTICS FOR PAIRED COMPARISON

ABSTRACT. This article provides an overview of recent results on lexicographic, linear, and Bayesian models for paired comparison from a cognitive psychology perspective. Within each class, we distinguish subclasses according to the computational complexity required for parameter setting. We identify the optimal model in each class, where optimality is defined with respect to performance when fitting known data. Although not optimal when fitting data, simple models can be astonishingly accurate when generalizing to new data. A simple heuristic belonging to the class of lexicographic models is Take The Best (Gigerenzer & Goldstein (1996) Psychol. Rev. 102: 684). It is more robust than other lexicographic strategies which use complex procedures to establish a cue hierarchy. In fact, it is robust due to its simplicity, not despite it. Similarly, Take The Best looks up only a fraction of the information that linear and Bayesian models require; yet it achieves performance comparable to that of models which integrate information. Due to its simplicity, frugality, and accuracy, Take The Best is a plausible candidate for a psychological model in the tradition of bounded rationality. We review empirical evidence showing the descriptive validity of fast and frugal heuristics.

KEY WORDS: models, lexicographic, linear, Bayesian.

1. INTRODUCTION

In most everyday situations humans have to make decisions quickly based on scarce information. The constraints of limited time, knowledge, and computational capacities have to be taken into account when modeling real-time decision making. If one must immediately respond to incoming information, the decision mechanism needs to be fast. One way to achieve speed is to be frugal, that is, to use (and to be able to work with) only little information. Yet, if fast and frugal decision rules did not work accurately they would, in an evolutionary time scale, be replaced by fitter ones. Thus, from such an evolutionary viewpoint persistent strategies need to be accurate, that is, fit, in order to be considered plausible models of actual hu-

Theory and Decision **52:** 29–71, 2002.

man reasoning. It is the claim of Gigerenzer, Todd, and the ABC Research Group (1999) that the mind is equipped with an *adaptive toolbox* of fast, frugal, and fit heuristics. In their approach the term heuristic has to be interpreted in its classical, historical sense, that is as a simple, useful procedure for solving problems. It should be noted that the term has recently been used to denote a rule of thumb used by humans which often fails to achieve the normative solution (Kahneman et al., 1982). In contrast, Gigerenzer et al. emphasized the fitness of heuristics proving that they do not fall too far behind the normative benchmarks.

Heuristics have not only been studied in cognitive psychology. Other fields, such as artificial intelligence and machine learning, have sensed the necessity of investigating simple strategies for problem solving due to the intractability of normative solutions. One example is simple algorithms in machine learning, which can perform highly accurate classifications. For instance, Holte (1993) showed that very simple classification rules, based on the use of only one—namely the best—dimension, perform surprisingly well compared to neural networks for classification. Another class of simple heuristics in artificial intelligence are decision lists, introduced by Rivest (1996). These heuristics are simple both in their application and with respect to the procedure for constructing them. This is different for the Classification and Regression Trees (CARTs) proposed by Breiman et al. (1993) which can be easily applied but may require enormous computational effort for their construction.

In the present paper we will focus on lexicographic strategies for paired comparison. For performing comparisons, speed can be achieved by searching for information with a simple search rule, stopping as early as possible, guaranteeing frugality, and making the inference with a simple decision rule. As we shall see, such fast and frugal heuristics can be astonishingly accurate and robust, and therefore, fit. They owe their fitness to their ecological rationality, that is, to the way in which they exploit the structure of their task environments.

These heuristics are simple to apply and do not require the assistance of a computer, not even paper-and-pencil calculations. The only computational effort involved concerns the order in which information is searched for. In Section 3 we will illustrate the simpli-

city and robustness of simple search rules, on the one hand, and the intractability and fragility of optimal orderings, on the other. Lexicographic strategies with simple search rules are not only tractable, as we will show for specific cases, but they also tend to be robust, that is, they generalize well to new, unknown data. We will use two important classes of benchmarks to evaluate the performance of these fast and frugal lexicographic strategies: linear models (Section 4) and Bayesian networks (Section 5). We commence by describing the type of task and give a brief overview of standard approaches to solve it.

2. APPROACHES TO THE COMPARISON TASK

In general terms, the task is to infer which object, A or B, has the higher value on a numerical criterion. This inference is to be made on the basis of cues. Here, for simplicity, we assume that all cue values are known for all objects and we restrict ourselves to binary cues, valued at either 1 or 0. From a formal point of view, the task is that of categorization: A pair (A, B) is to be categorized as $X_A > X_B$ or $X_B > X_A$ (where X denotes the criterion), based on cue information. Examples of such a task are 'choose which of two German cities has the larger population', or 'choose which of two stocks will yield a higher return'.

The matrix of all objects of the reference class, from which A and B have been taken, and of the cue values which describe these objects constitute what we call an environment. A cue profile of an object is the vector formed by all the cue values of the object. Thus, an environment consists of a set of objects, where each object is characterized by a criterion value and a cue profile. As a concrete, real-world example we will consider the task originally investigated by Gigerenzer and Goldstein (1996), where pairs of German cities with more than $100,000$ inhabitants were compared as to which one has a larger population. There were nine cues, such as whether the city is a state capital or whether it has a soccer team in the national league. Information on these cues is useful since cities which, for example, have a soccer team in the major league tend to have more inhabitants than those which do not. The performance of an inference mechanism is determined as the percentage of correct

TABLE I

Three approaches to the task of inferring which of the two alternatives scores higher on a numerical criterion. The variants within each of the three approaches vary with respect to their complexity in finding the parameters relevant to the model.

Complexity	Lexicographic approach	Linear approach	Bayesian approach
Minimal	Minimalist*	Dawes's Rule	Minimal Bayes
Simple	Take The Best	Franklin's Rule	Naïve Bayes (Idiot Bayes)
Sophisticated	Cue ordering based on conditional validity	Multiple regression at the object level	Friedman's network
Optimal	Optimal Cue Ordering	Logistic regression at the comparison level	Profile memorization

Note: *Although Minimalist shares many features of lexicographic strategies, it does not belong to this class of strategies since it does not have a prespecified cue ordering.

inferences in the complete paired comparison task, that is, across all possible combinations of objects of the environment.

How can one infer which of two objects, for example, city A with cue profile (100101010) and city B with cue profile (011000011), scores higher on the established criterion? Table 1 illustrates three traditional approaches to this task: lexicographic, linear, and Bayesian which are the focus of this paper (there are other possible approaches, e.g., neural nets and non-linear regression).

2.1. *Lexicographic strategies for comparison*

A comparison task we are all familiar with is looking up a word in the dictionary. We compare the entry we accidentally find with the target word and have to determine whether the target word is before or after. We do not have to read the whole word; rather, we compare corresponding characters with the target word from the beginning until we find a character that discriminates. The equivalent can be carried out when comparing the cue profiles of two objects: Compare cue values in a specified order and make an inference as soon as a discriminating cue is found. One essential feature of a lex-

icographic strategy is its search rule, that is, the order in which the cues are checked. Methods to fix this order vary in complexity. We distinguish those that (1) require minimal effort, (2) are simple, or (3) sophisticated, and those that (4) strive for the optimum (Table 1). We will introduce and evaluate some of these methods in Section 3.

2.2. *Linear models for comparison*

Linear models are well established in decision theory (Cooksey, 1996; Kurz and Martignon, 1999). In contrast to lexicographic strategies, which may stop acquiring further information depending on the information already checked, linear models collect and process all information which is required. While a lexicographic strategy is characterized by its cue order, a linear model is characterized by its set of cue weights. Methods to set up these weights vary in complexity. There is a natural analogy between methods to set up weights for a linear model and search rules for lexicographic strategies (see Table 1). We will discuss the relation between lexicographic strategies and linear models in Section 4.

2.3. *Bayesian networks for comparison*

A Bayesian facing the comparison task asks the question 'Given a pair of objects, A and B, and their cue profiles, (100101010) and (011000011), what is the probability that A has a larger criterion value than B?' The Bayesian uses all the cue information which the network requires. The network is characterized by the links between its nodes, where these nodes represent cues and a target criterion. A link between two cues represents conditional dependence, while the absence of a link denotes conditional independence. Here again, methods for determining the network vary in complexity (see Table 1) and will be explained in Section 5.

3. SEARCH RULES FOR LEXICOGRAPHIC STRATEGIES

Here, we introduce methods to set up the orderings for lexicographic comparison (Sections 3.1–3.5) and then evaluate them (Section 3.6–3.8).

3.1. *A simple search rule: Take The Best*

An attractive way of searching for cue information is checking cue values along the order established by the cue validities (Gigerenzer et al., 1991). The validity of a cue is its predictive accuracy. This simple search rule defines a lexicographic strategy which has become known as the Take The Best heuristic (Gigerenzer and Goldstein, 1996). Take The Best can be described as follows (excluding the recognition principle, which plays no role here, since we assume that all objects are recognized):

Pre-processing phase. Compute the validities defined by

$$v = \frac{R}{R + W} \tag{1}$$

for each cue, where R is the number of right (correct) inferences, and W the number of wrong (incorrect) inferences based on that cue alone, when one object has the value 1 and the other has the value 0. Cues with a validity of 0.5 are neutral and are, taken by themselves, of no assistance for the comparison task. Cues with a validity of more than 0.5 are beneficial, and so are cues with a validity of less than 0.5. For convenience, we will invert cues with validity less than 0.5, that is, change each 1 to a 0 and each 0 to a 1. After having inverted cues where necessary, we rank all cues according to their validity.

Step 1. Search rule: Pick the cue with the highest validity and look up the cue values of the two objects.

Step 2. Stopping rule: If one object has a cue value of one (‘1’) and the other has a value of zero (‘0’) then stop the search. Otherwise, pick the cue with the highest validity among the remaining ones and return to Step 1. (This simple stopping rule needs to be refined if binary cues can have unknown values (Lages et al., 1999), or if continuous cues are considered (Slegers et al., 2000).)

Step 3. Decision rule: If one object has a cue value of one (‘1’) and the other has a value of zero (‘0’) predict that the object with the cue value of one (‘1’) has the higher value on the criterion. If no cue discriminates, guess.

Thus, Take The Best simply looks up cues in the order of their validity. Note, that to compute this validity all pairs of objects have to be considered. Another, quite different but practical way of computing the ecological validity is the following:

$$v = \frac{S_0 - \dfrac{N_0(N_0 - 1)}{2}}{N_0 N_1} \tag{2}$$

where S_0 denotes the sum of all ranks of objects (ordered according to decreasing criterion) with a 0 entry, N_0 is the number of 0 entries, and N_1 the number of 1 entries. The advantage of (2) is that cue validity can be computed without generating all pairs. Note, that the numerator corresponds to the well-known U-value for the Mann–Whitney test and the denominator corresponds to $R + W$ (the equivalence between (1) and (2) is shown in the Appendix). Also note that the well-known Goodman–Kruskal rank correlation γ defined by

$$\gamma = \frac{R - W}{R + W} \tag{3}$$

is a positive rescaling of v. In fact, a simple calculation shows that $\gamma = 2v - 1$. Thus, both notions of validity, v and γ, establish the same hierarchy when used as a criterion to order cues, and both can be used as alternatives.

3.2. *Other simple search rules*

The ecological validity v is one possible criterion for ordering cues. Another approach to defining the validity of a cue is to view the performance of Take The Best when only the cue in question can be used. Each time the cue does not discriminate between two objects, we flip a coin. The performance of a Take-The-Best-type algorithm based only on this cue is what we call the *success* (s) of the cue.

$$s = \frac{R + 0.5(P - R - W)}{P} \tag{4}$$

where P is the total number of pairs. The ecological validity v is related to success: It is the probability of the success of a cue, conditional on discrimination.

Another relevant candidate definition for validity is Kendall's τ, which is given by a slight modification of γ, namely,

$$\tau = \frac{R - W}{\sqrt{P(R + W)}} \tag{5}$$

where P again denotes the number of pairs, R the number of right inferences, and W the number of wrong inferences (for details on γ and τ, see Gigerenzer, 1981).

3.3. *Random search: Minimalist*

Minimalist (Gigerenzer et al., 1999) is quite similar to Take The Best. The only difference is in Step 1, which now reads: 'Pick a cue randomly (without replacement) and look up the cue values of the two objects'. Thus, Minimalist does not have to know the validity of the cues, but only their direction (i.e., whether the validities are above or below 0.5, before the cue values are—eventually—inverted). Since the cues are not checked in a specified but in a random order, Minimalist is—strictly speaking—not a lexicographic strategy. What is common for both heuristics is that search for information is stopped as soon as a cue is found on which the two alternatives differ. This simple stopping rule demands no cost–benefit considerations: Its motto is 'Take the best reason and ignore the rest', in one case, and 'Take any valid reason and ignore the rest', in the other.

3.4. *A sophisticated search rule: Conditional validity*

Take The Best's rule for ranking cues is elementary: It inverts 'bad' cues in its preprocessing phase and orders them according to their validities. This procedure is simple, but does it lead to the optimal ordering? There are reasons to be suspicious that other hierarchies may lead to better performance. Note, that the hierarchy is determined by computing the validity for each cue in the complete set of all possible pairs, while each cue leads to inferences only on the subset of pairs left undiscriminated by preceding cues (i.e., pairs on which each of the preceding cues has coincident values). A sophisticated method to determine the cue hierarchy for a lexicographic strategy which eliminates this discrepancy is to turn to *conditional validity (cv)*, which is defined as follows. It is computed for each cue just

on the set of pairs not discriminated by the cues already used. The first cue used by this type of search is the most valid, as for Take The Best. The second cue is the most valid on the set of pairs that the first cue did not discriminate, and so on; if the validity of a cue on the remaining pairs turns out to be below 0.5, the values of this cue are inverted by changing 1s into 0s and vice versa. Since the reference class, in which the conditional validity is determined, coincides with the set of pairs on which the cue with the highest conditional validity is checked, the following is a straightforward result.

THEOREM 1 (Conditional validity). *The accuracy of a lexicographic strategy with conditional validity is larger than (or equal to) that of Take The Best.*

A lexicographic strategy which uses conditional validity to establish the cue hierarchy is reminiscent of Classification and Regression Trees (CARTs; Breiman et al., 1993), that also determine their branches sequentially by optimizing performance at each step.

3.5. *Optimal ordering*

Does using conditional validity already yield the optimal ordering for lexicographic comparison, that is, the ordering which achieves the highest performance? We ask the more general question: What is the optimal ordering and how can it be obtained? It can be obtained, for instance, by exhaustively searching through all possible permutations of the given cues (and their inverses) and by computing the performance obtained for each ranking. This is an extremely laborious task. Is there a simpler way to find the optimal ranking? By simpler we mean shorter, involving fewer steps (such as using conditional validity). To answer this question we first have to rigorously define simplicity/complexity of an algorithm. In theoretical computer science the complexity of a sequential algorithm can be defined in terms of the time required to perform all necessary operations.

One possibility is to count the number of operational steps in an algorithm, express this number as a function of the parameters involved in the algorithm, and determine the order of complexity of this function. The popular criterion for order of complexity is denoted by O( ) (usually called the Landau symbol) and is defined as

follows: Given two functions $F(n)$ and $G(n)$ defined on the natural numbers, we state that F is—at most—of the order of G, and write $F = \mathrm{O}(G)$, if a constant K exists such that

$$\frac{F(n)}{G(n)} < K \tag{6}$$

for all natural numbers n. Thus, for instance, the function $F(n) = 3n + 1$ is of the order of $G(n) = n^2$, but not vice versa. Every polynomial in n, that is, every linear combination of powers of n, is of the order of its highest power of n. Since what is being counted is the number of steps in a sequential process, it is common to view the resulting O( ) criterion as the time complexity of the algorithm (more precisely, the amount of time needed by the algorithm to calculate the solutions), where n denotes the length of the given input.

A problem is said to be solvable in polynomial time if it is of the order of some polynomial in n. In contrast, if there is no machine, be it deterministic or nondeterministic (i.e., allowing for steps that involve stochastic decisions), which solves the problem in polynomial time, it is called NP (Garey and Johnson, 1979). In particular, if this process were to be simulated by a deterministic Turing machine it would require more than polynomial time. Informally, a computational problem is said to be NP-hard if its being in P implies that P = NP. NP-hardness results are in general obtained via so-called reductions. The reduction of a problem A to another problem B implies that any algorithm which solves B efficiently can be transformed into an efficient algorithm which solves A. A now widely used technique to prove NP-hardness of a problem works as follows: Choose a suitable problem in NP that is known to be NP-hard and find a reduction to the problem whose NP-hardness is to be established. This necessarily requires that the class NP contains NP-hard problems. By now, computer scientists have discovered a class of several hundred NP-hard problems in NP (Garey and Johnson, 1979). It has become a common trend in computer science to assume that P $\neq$ NP. Thus, although still standing on hypothetical grounds, the notion of NP-hardness has become an equivalent to intractability with relevant consequences for the use of algorithms in practical applications.

Now, let us return to the problem of finding an optimal ranking of cues. Given n cues, what is the complexity of finding an optimal

ranking? Searching across all permutations of n objects requires $n!$ steps and is, hence, of an order larger than any fixed power of n. Recently it has been proven that there are no essentially simpler strategies to find the optimal ranking of n cues (Martignon and Schmitt, 1999).

THEOREM 2 (Intractability of optimal ordering search). *The problem of determining the optimal ordering of binary cues for lexicographic comparison is NP-hard.*

We sketch the concept of the proof without going into the formal details. It turns out that a famous problem called the Vertex Cover Problem is suitable for proving the NP-hardness of the optimal ordering search of binary cues. The Vertex Cover Problem stems from graph theory and consists of finding a set of nodes such that each edge is 'covered', that is, touched by some node in the set, the cover and the cardinality of this set is minimal. The proof operates by displaying how to construct efficiently, from a given graph, a set of binary cues such that the cardinality of the smallest cover in the graph is a simple function of the number of correct inferences in the optimal cue ranking. Thus, a reduction is established by showing that detecting the optimal ranking of binary cues is at least as difficult as finding a minimum cardinality vertex cover in a graph.

A consequence of Theorem 2 is that conditional validity does not necessarily yield the optimal ordering. The question, whether in a *specific* environment conditional validity yields the optimal ordering can only be answered empirically, that is, by finding the optimal ordering through exhaustive search and comparing this ordering to the one implied by conditional validity.

3.6. *Performance of lexicographic strategies in the fitting task*

How does the performance of lexicographic strategies depend upon their search principles? We answer this question for the task of comparing German cities, using the environment shown in the Appendix of Gigerenzer and Goldstein (1996). Figure 1 shows the accuracy distribution of all possible orderings. The optimal ordering for this comparison task is, by defintion, the one which achieves the highest performance (75.8%, the right extreme in this distribution). The mean of the distribution (70.0%) corresponds to the expected per-

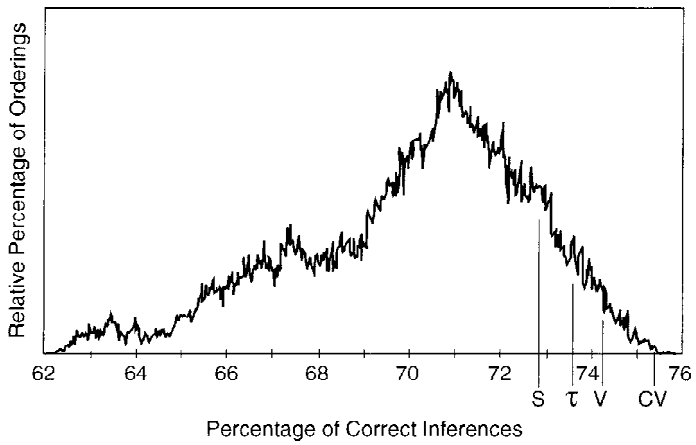

Figure 1. Distribution of performances for the 362 880 possible cue orderings in the German city environment. The mean of the distribution corresponds to the expected performance of Minimalist (random ordering). The performance of lexicographic strategies, which search for cues according to ecological validity (i.e., Take The Best), success validity, Kendall's τ, and conditional validity are denoted by v, s, τ, and cv, respectively. (Reprinted with permission of Oxford University Press.)

formance of Minimalist. All other search principles have a performance between these two values: Conditional validity (cv) achieves 75.6%, ecological validity v, as used by Take The Best, achieves 74.2%; Kendall's τ achieves 73.5%; and Success (s) achieves 72.7%. Only 1.8% of the orderings allow a higher performance than the ordering of Take The Best. Thus, the search principle of Take The Best achieves a satisfiying performance, in that improving this performance involves paying too high a price in terms of computational complexity; to determine the optimal ordering for the German city environment takes a UNIX machine two days!

3.7. *Deviations from Take The Best's cue hierarchy*

Take The Best fixes its cue hierarchy by ranking cues according to their validities. Unlike a computer, humans may rank the cues in a given environment without *computing* their validities. Alternativley, they may *estimate* validities, that is, relative frequencies, to establish the order. Literature on automatic processing of frequencies (Hasher and Zacks, 1984) suggests that these estimates will be quite accurate. However, it is questionable whether a human's cue order

perfectly matches Take The Best's order. The question is whether small violations of Take The Best's ordering cause dramatic changes in performance. We deal with this question in the specific case of the German city environment. First, as a measure of the distance from Take The Best's ordering to any other ordering, we use the minimal number of transpositions required to obtain this new ordering from Take The Best's original ordering. For instance, if Take The Best's ordering is 1,2,3,4,5,6,7,8,9, the distance to 2,1,3,4,5,6,9,7,8 is three, since we have to perform three transpositions. (one by changing the order of Cue_1 and Cue_2 and two others by moving Cue_9 from the end to its new position). The distances across all $362,880$ possible orderings of the nine cues range from 0 to 36. Figure 2 displays the average performance — across all orderings with the same distance from Take The Best's ordering — as a function of this distance. Small violations against Take The Best's hierarchy have only negligible effects on performance. For instance, across the 628 orderings, with a distance of five to Take The Best's ordering, the average performance (73.6%) was only 0.6 percentage points below that of Take The Best. The expected distance from Take The Best's ordering to Minimalist's ordering is 18, which corresponds to an average performance of 4.2 percentage points below Take The Best.

3.8. *Performance of lexicographic strategies in a generalization task*

It is one thing to fit known data, as we have done so far. It is another to train a model on a set of data and test it on another, unknown set. If both these sets are drawn from a larger, homogeneous one then the capacity of the model to generalize well depends on its ability to extract relevant structure without fitting noise. In cognitive psychology, models that generalize well from the training set to the test set are called robust (Dawes, 1979). The term 'robust' has slightly different connotations in statistics and in mathematics (more precisely, in the theory of dynamical systems) but it always refers to stability of a model with respect to specific changes. A model is said to overfit the data if there is another model which performs less well on the training set but better on the test set.

How robust are the search rules introduced above? To answer this question, we need to compare the accuracy when fitting known

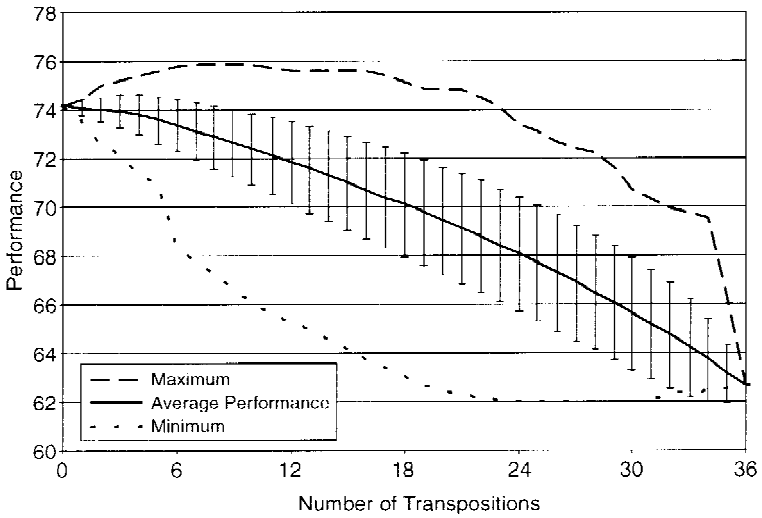

Figure 2. Performance of lexicographic strategies in the German city environment as a function of the minimal number of inversions which were necessary to match Take The Best's cue hierarchy. Dashed lines denote the maximum and minimum performance of all orderings for a given number of necessary inversions; bars denote the standard deviations.

data with the accuracy obtained when generalizing to unknown data within a homogeneous population — from a subset to either the whole population or to another subset. Here, we check the robustness of search principles by taking a random subset of half of the German cities, for which we determine the cue orderings, checking the performance first in the subset (i.e., the training set), and then in the other half of the cities (i.e., the test set). We repeated this procedure, known as cross-validation, 100 times for randomly chosen subsets. The results are shown in Figure 3. As expected, for all orderings, the performance in the test set is lower than that in the training set. Surprisingly, Take The Best uses the most robust search principle. The complex search principles (optimal ordering and *cv*) dropped the most, indicating that they fitted noise in the training set.

Let us briefly summarize this section. Take The Best has been introduced as a fast and frugal lexicographic strategy. In its decision phase this strategy is fast since it does not involve any computation and can, thus, easily be applied. It is frugal since it uses only a fraction of the available information. Moreover, the computation necessary in the preprocessing phase is also fast and frugal. The cue

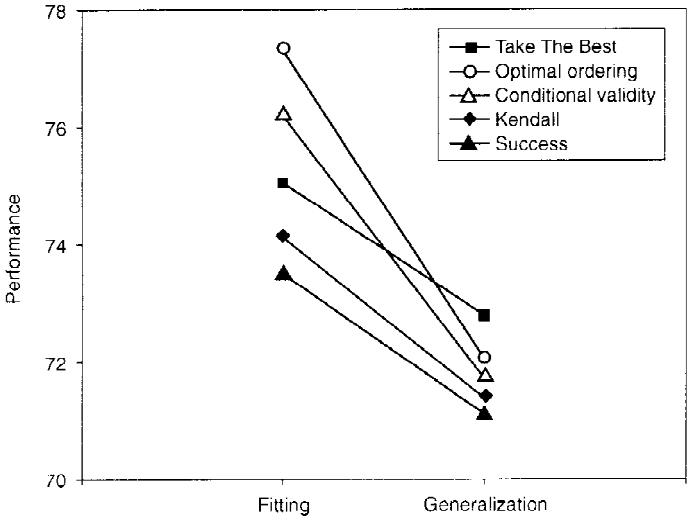

Figure 3. Robustness of lexicographic strategies with different principles of search in the German city environment. The training set consisted of half of the cities randomly drawn from the environment, the test set of the remaining half. Performance is averaged across 100 trials.

validities, which determine Take The Best's cue hierarchy, can easily be calculated for each cue without considering relations between cues (as is the case, for instance, for conditional validity), and sorting the cues according to their validity is simple. The fact that its search rule is more robust than others, which are more subtle, leads us to the conclusion that Take The Best is the 'best'—at least within the class of lexicographic strategies. We next turn to a comparison with linear models.

4. BENCHMARKS IN THE CLASS OF LINEAR MODELS

4.1. *Variants of linear models*

Assessing the performance of Take The Best requires a comparison with appropriate benchmarks. Such benchmarks can be found, for instance, in the class of linear models, which have a longstanding tradition in decision making. For the comparison task a linear model can be built on two levels, the object level and the comparison level. At the object level one computes a score for each object—by weighting the cue values and summing them in linear fashion—

and inferring that the object with the larger score is also the one with the larger criterion value. This is the implementation used by Gigerenzer and Goldstein (1996), and Gigerenzer et al., (1999). At the comparison level, in contrast, one computes a score for a pair of objects and makes the inference based on a comparison between this score and a threshold value.

The most popular linear model is multiple linear regression, whose weights are calculated to minimize the average square distances between predictions and criterion values. Determining this vector of cue weights is simple for users of modern statistical packages, however, the underlying mathematics is sophisticated, involving the inversion of the cue intercorrelation matrix. Does this procedure yield the optimum in the linear world? When the task is to predict the criterion values of known objects, and when the optimum is defined as the minimum of the average square distances between predictions and criterion values, multiple regression is, by definition, the optimum. For the comparison task under common distributional assumptions, however, the optimum is logistic regression at the comparison level. From a psychological point of view, linear models at the object level are more plausible than those at the comparison level, therefore, the latter are not further pursued in the present paper.

Dawes and Corrigan (1974) have shown that using unit weights ('1' if the correlation between cue and criterion is positive; '-1' if it is negative), and even random weights (i.e., weights which are randomly chosen except for sign), yields astonishingly accurate results, in particular when generalizing to new data. Following Gigerenzer, Todd, and the ABC Research Group (1999), we call a simple linear model with *binary* unit weights *Dawes's Rule*. Dawes's Rule corresponds to the Minimalist (Table 1) in the sense that all it requires to know are the directions in which the cues point. Whereas Dawes and Corrigan (1974) used correlation coefficients to determine this direction, we use cue validities (which are simpler to calculate) to determine the sign of the unit weights ('1' if $v > 0.5$, '-1' if $v < 0.5$). Thus, the scores for Dawes's Rule are the number of 1s minus the number of 0s in the cue profile. If all cue values are known, as we assume in this paper, Dawes's Rule is equivalent to comparing the scores obtained by counting the number of 1s in each

profile. Twice the number of 1s minus the number of objects is equal to the number of 1s minus number of 0s. Therefore, the orderings defined by both scores coincide.

A slightly more complex linear model than Dawes's Rule, yet simpler than regression, has been named *Franklin's Rule* (Gigerenzer et al., 1999). Franklin's Rule uses the Goodman-Kruskal validities of the cues as their weights, and it is thus related with Take The Best, which uses validities to establish its cue ordering.

4.2. *Performance of linear models*

Figure 4 shows the performance of Take The Best, Minimalist, and three linear models in 20 data sets collected from different real-world domains (Czerlinski et al., 1999). As can be seen, Take The Best is well able to compete with these linear models, performing only four percentage points below multiple regression. The next step is to compare the algorithms in a generalization task. Czerlinski et al. (1999) also performed cross validation on these 20 data sets, by dividing the sets in half (determined at random), computing the required parameters (for the lexicographic strategies, cue direction and order; for the linear models, cue weights) on one half, and testing the algorithms with these parameters in the remaining half. The results, shown in Figure 4 are amazing! When cross-validated, simple heuristics such as Take The Best even outperform sophisticated linear models. Take The Best is robust due to its simplicity, not despite it. Its smart simplicity protects Take The Best from the danger of overfitting and 'squeezing' spurious information out of the data.

4.3. *Take The Best and linear models: The case of noncompensatory information*

Take The Best is based on one-reason decision making: The decision which of two objects scores higher on the criterion is made solely on the basis of the most valid cue that discriminates between the two objects. The decision may be wrong, yet none of the remaining cues, nor any combination of them, will change it. In other words, Take The Best is a noncompensatory strategy. Such a strategy works best if the environment has a similar structure, where each cue is more important than any combination of less valid cues.

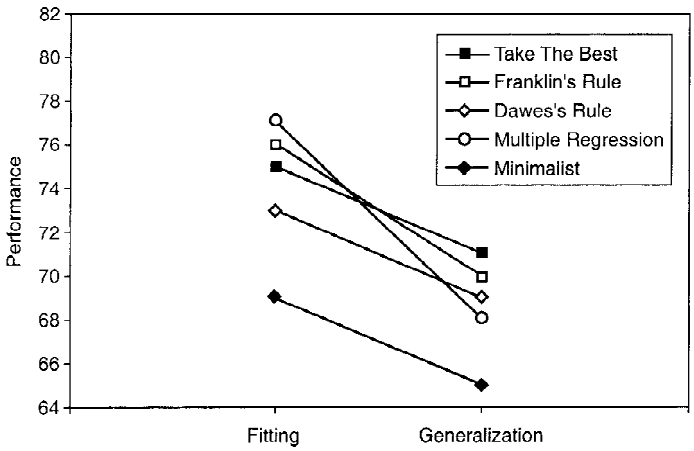

Figure 4. Performance of Take The Best, Minimalist, and three linear models (Dawes's Rule, Franklin's Rule, multiple regression) averaged across 20 different data sets. In the generalization task, 50% of the objects in the data set were chosen, at random, 1000 times and the model obtained on the training set was then tested on the remaining 50%. All results except for Franklin's Rule were taken from Czerlinski et al. (1999).

If cue weights are noncompensatory, then linearly combining cues yields the same performance as processing cues in lexicographic fashion, one cue at a time (see Theorem 3, below). This can be illustrated in the task of comparing two numbers written in the decimal system, that is, as linear combinations of powers of 10. For instance, to establish that 357 is larger than 318, we can check the value of the first digits on the left, and, since they coincide, move to the second digits, which differ. At this moment we make the decision that the number with 5 as second digit is larger than the number with 1. (Similar to the decimal system, any other base system has noncompensatory weights and would, thus, also allow for this type of strategy.)

We now define, in general terms, what a noncompensatory strategy is. Consider an ordered set of M binary cues, $C_1, \ldots, C_M$. Loosely speaking, these cues are noncompensatory for a given strategy if every cue C_j outweighs any possible combination of cues after C_j, that is, C_{j+1} to C_M. In the special case of a weighted linear model with a set of weights $W = \{w_1, w_2, \ldots, w_M\}$, a strategy is noncompensatory if for each $1 \leq j \leq M$ we have $w_j > \sum_{k>j} w_k$. In words, a linear model is noncompensatory if, for a given ordering

of the weights, each weight is larger than the sum of all weights to come. A simple example is the set {1, 1/2, 1/4, 1/8, 1/16}.

A linear model with a noncompensatory set of weights results in making identical inferences as Take The Best. The converse is also true: The performance—but not the process!—of a lexicographic strategy is identical to that of a linear model with noncompensatory weights.

THEOREM 3 (Noncompensatory information). *The performance of Take The Best is equivalent to that of a linear model with a noncompensatory set of weights (decreasing in the same order of Take The Best's hierarchy). If an environment consists of cues, which for a specified order are noncompensatory for a given weighted linear model, this model cannot outperform the faster and more frugal Take The Best if that order coincides with the decreasing order of validities (an analytical proof is given in the Appendix).*

Liberally speaking, Take The Best embodies a noncompensatory structure and, if the environment has the same structure, then there is a fit. The degree to which this fit exists contributes to the ecological rationality of Take The Best. Three of the 20 data sets of Figure 4 have noncompensatory regression weights decreasing in the same order as do the validities of the cues; furthermore, as stated by Theorem 3, the performance of regression and that of Take The Best are identical. If the fit is not perfect, but approximate, then Take The Best will still be about as accurate as the corresponding linear model.

In a nutshell, Take The Best 'bets' that the cues in the environment are noncompensatory, whereas Dawes's Rule bets that they are equally important for making an inference. Multiple regression, in contrast, does not make blind bets, but computes its parameters to fit the structure. The price for this is more computation and, as could be seen in Figure 4, less robustness.

4.4. *Scarce information*

If information is skewed, a lexicographic strategy can be used as a shortcut for a linear model, since both are equivalent in performance. The next two theorems specify conditions, under which Take The Best and Dawes's Rule differ in performance (Martignon and

Hoffrage, 1999). Theorem 4 considers the difference between these heuristics with respect to the (size of the) matrix of M binary cues and N objects. Environments are said to contain *scarce information* if $M \leq \log_2 N$. The rationale behind this definition is that one requires at most $\log_2 N$ yes–no questions to identify one out of N objects. More generally, according to information theory (Shannon, 1948), a class of N objects contains $\log_2 N$ bits of information. The following result has been shown to be true for small environments with up to 2^7 objects by exhaustive counting (with the help of a Cray computer; obtained by Michael Schmitt).

THEOREM 4 (Scarce information). *In the majority of small environments with scarce information, Take The Best is more accurate than Dawes's Rule.*

Although we do not have a formal proof for environments with a larger numbers of objects, simulations suggest that Theorem 4 can be true for environments with more than 1024 objects. An intuitive explanation of this result is that when based on a large number of cues, Dawes's Rule can compensate for possible errors in the first ones and has a high discrimination rate. With scarce information, these advantages are lost: Dawes's Rule cannot really exploit compensation and is forced to make numerous guesses.

4.5. *Abundant information*

The next theorem considers the opposite case, that is, environments with abundant rather than scarce information. In general, adding information to a scarce environment will do little for Take The Best, while it can compensate for mistakes Dawes's Rule makes when based only on the first cues. Observe that Dawes's Rule does not discriminate between profiles (110) and (011), whereas Take The Best does. What if more and more information, that is, more and more cues are added to a given number of objects? We state that an environment provides *abundant information* when all possible uncertain, yet valid, cues are present. In an environment with N objects and binary cues, the number of possible cues is the number of different 1–0 sequences of length N. Note, that the expression 'all possible uncertain, yet valid, cues' does not refer to all possible real-world cues but to the different 1–0 sequences. Whereas the number

of real-world cues is infinite (since different real-world cues may have the same value for each object), the number of different 1–0 sequences is finite. The following result is true for environments with five or more objects (the analytical proof is given in the Appendix).

THEOREM 5 (Abundant information). *When information in an environment is abundant, Dawes's Rule makes a correct inference for each possible pair of objects. The same is true of Franklin's Rule.*

In contrast, the more frugal Take The Best cannot achieve perfection in such environments, because its errors cannot be compensated by later cues.

4.6. *Cue validities*

In order to see how the performance of Take The Best (as compared to that of Dawes's Rule and Franklin's Rule with Goodman-Kruskal validities) depends on the validities we artificially generated 10 000 environments, each with 16 objects and four binary cues. The cue values have been randomly generated. If the validity of a cue was below 0.5, the cue values were flipped, such that all validities were above 0.5 subsequent to this procedure. We subdivided these environments according to the average validity of their cues, taking the median of these averages as the split point. Across all 5000 environments with mean validity above this split point, the performance of Take The Best, Franklin's Rule, and Dawes's Rule were 69.8, 70.0, and 66.9, respectively. Across those 5000 environments with mean validity below the split point, the performance was 61.8, 62.0, and 59.7, respectively. Thus, the difference between the performance of Take The Best's and Franklin's Rule (Dawes's Rule) was mostly unaffected by mean validity: it was -0.2 (2.9) across all environments with higher mean validities and -0.1 (2.2) across all with lower mean validities. The correlation coefficients between mean validity of an environment and the performance of the strategies in the particular environment was quite high (in each case at least 0.8), however, all strategies were effected in approximately the same manner. The correlation coefficients between the mean validity and the difference of Take The Best's performance minus Franklin's (Dawes's) Rule's performance was 0.03 (0.12). Thus, the average validity appears to play a negligible role in the comparison of Take

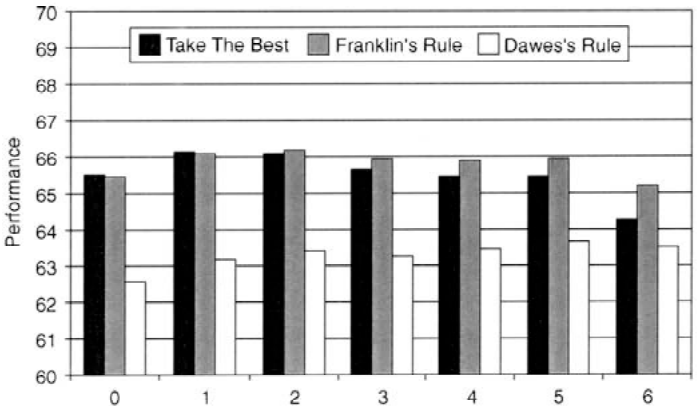

Figure 5. Performance of Take The Best, Franklin's Rule, and Dawes's Rule across all environments with the same number of positve cue intercorrelations. The environments consisted of 16 objects and four cues, for each environment ($n = 10\,000$) cue values have been randomly generated – if the validity of a cue was below 0.5, the cue values have been flipped.

The Best, Franklin's Rule, and Dawes's Rule with respect to their performance.

4.7. *Cue intercorrelations*

For the same 10 000 environments we also computed how many of the six intercorrelations between the four cues turned out negative. Figure 5 displays the performance of Take The Best and the two linear models averaged across environments with the same number of negative intercorrelations (there were 663, 1726, 2759, 2909, 1534, 384, 25 environments with 0, 1, 2, 3, 4, 5, and 6 negative intercorrelations, respectively). On average, Take The Best outperformed both Franklin's Rule and Dawes's Rule when there was no or only one negative intercorrelation between the four cues, whereas Franklin's Rule could profit in environments with two or more negative intercorrelations. Note, that although Dawes's Rule showed, on average, the worst performance in each category, it was the only heuristic that showed a positive correlation between performance and number of negative cue-intercorrelations.

To summarize, we analyzed four information structures—skewness (noncompensatory versus compensatory), amount of information (scarce versus abundant), average validity (high versus low) and cue-intercorrelations (ranging from all positive to all negative)—and

examined how Take The Best performs in each context as compared with two linear models. Apart from average validity, each of these structures had an impact on the performance of Take The Best (as compared to that of Franklin's Rule and Dawes's Rule), demonstrating some aspects of Take The Best's ecological rationality. The fit between fast and frugal heuristics and the structure of environments explains why there need not always be a trade-off between simplicity and accuracy—a heuristic can have it both ways.

5. THE BAYESIAN TOOLBOX AND THE COMPARISON TASK

This section is devoted to the comparison between fast and frugal heuristics and the Bayesian benchmarks (Martignon and Laskey, 1999). The best known theory of unbounded rationality is Bayesianism, according to which a rational agent chooses the action that maximizes expected payoff, where the expected payoff is calculated from the probabilities of *all* payoffs for each possible action in every possible environment. The problem is that expected payoffs are expensive to calculate. One can envisage a Bayesian justification for acting and deciding according to simple rules. In fact, a Bayesian could agree that acting rationally is to act in accordance with a set of simple rules, and a set of rules is shown to be correct by showing that it achieves a utility which is almost as high as the maximum. The justification of the rules is still computationally expensive, but it is carried out 'offline' (i.e., it corresponds to the preprocessing phase), and therefore, allows decision making in real time to be fast and frugal. If fast and frugal heuristics perform well even when compared to Bayesian benchmarks, then even the Bayesian decision maker may deliberately decide not to be a Bayesian.

Although Bayesianism represents today's leading trend in statistical inference, modern experimental psychology challenges the view of a rational person as probabilist in a flurry of work documenting the ways in which actual human reasoning differs from the probabilistic norm. These deviations have been named by many 'cognitive illusions', thus claiming that unaided human reasoning is riddled with fallacies (e.g., Tversky and Kahneman, 1974). However, Gigerenzer and Hoffrage (1995) have shown that being or not being a Bayesian is a consequence of how information is presented for a

given task. Assume the following categorization task is to be solved. We must infer whether or not a patient has breast cancer, given a positive mammogram. Experimental participants easily find the Bayesian solution if information is presented in a *frequency format*, while they have difficulties if information is presented in probabilities. Because the term 'frequency format' has been repeatedly misunderstood as frequencies of any kind, in recent publications we have, instead, adopted the (synonymous) term *natural frequencies* (e.g., Hoffrage et al., 2000). Natural frequencies result from the counting of individual cases observed in a natural environment. If we are interested in two variables—a cue and a criterion, such as symptom and disease—then natural frequencies are the frequencies of the four events in the 2×2 table. Thus, natural frequencies have not been normalized with respect to the base rates (the margins in the 2×2 table). Rather, they carry information concerning these base rates, and thereby, facilitate Bayesian computations.

Even if the task involves two cues (positive mammogram and positive ultrasound test) applying the Bayesian norm does not pose problems if information is presented in natural frequencies (Krauss et al., 1999). In addition, the generalization to three or more cues appears to be at hand. Yet for tasks with several cues, simply storing the probability distribution over all possible configurations often becomes impossible, even for computers. Reducing the complexity of the probability distribution over all possible configurations requires using a model. This model can again be selected with a Bayesian paradigm. Recently, the problem of model selection for complexity reduction has been tackled using smart implementations of Occam's Razor, which is a maxim recommending not exceeding the number of parameters beyond necessity. It is precisely this type of development that has created the vigorous renaissance of the Bayesian approach.

How does the Bayesian face the comparison task discussed in the above sections? She begins by expressing the comparison probabilistically: What are the chances that object A with cue profile, for example, (100101010) scores higher than object B with cue profile, for example, (011000011) on the established criterion? In symbols this is

$$p(X_A > X_B | A \equiv (100101010), B \equiv (011000011)) = \; ? \quad (7)$$

Here '≡' is used to signify 'has the cue profile'.

Observe that there are no essential differences between this task and the task of categorizing a patient as having or not having a disease based on a set of symptoms. To see the equivalence, let us rephrase the question as follows: What is the probability that the pair (A, B) belongs to the category $X_A > X_B$ (rather than to the category $X_B > X_A$, where X denotes the criterion), given that the first cue takes the value (1,0) on the pair, the second cue takes the value (0,1) on the pair, and so on.

5.1. *Profile Memorization Method*

When training set and test set coincide, the optimal performance in a complete paired comparison task can be achieved by the *Profile Memorization Method*. In the preprocessing phase, the Bayesian with perfect memory would memorize all cue profiles with their corresponding criterion values (thus resembling exemplar models for categorization). Then, she would use this information for inferring whether object A scores higher than object B. If a pair of profiles appears only once in the list of all possible pairs, she remembers which profile scored higher and makes that inference. If some cue profiles occur more than once and, consequently, several pairs of objects have the same pair of cue profiles, the probabilistic recipe is to count how often the first profile scores higher than the second one. Forced to a deterministic answer she will pick the profile that has the greatest chances of scoring higher. In environments where each cue profile appears exactly once, the Profile Memorization Method always achieves 100% correct inferences. If there are repeated profiles—and, therefore, also repeated pairs of profiles—this method will score less than 100% correct inferences and so will any other algorithm. When fitting the training set, there is no way to achieve a higher performance than using the Profile Memorization Method. On the other hand, this method is (usually) useless in a generalization task, since it is not able to deal with unknown cue profiles.

The Profile Memorization Method can be considered the one extreme in the family of Bayesian networks. A Bayesian network for our type of task considers pairs of objects (A, B) and the possible states of the cues, which are the four pairs of binary values (0,0),

(0,1), (1,0), (1,1) on pairs of objects. Although the computational complexity of the Profile Memorization Method is negligible, both in the preprocessing and in the decision phase, the memory load can be enormous. Storing the original distribution as it is, without decomposing it into simpler factors, corresponds to the fully connected network where each pair of nodes is connected both ways.

5.2. *Naïve Bayes (Idiot Bayes) and Minimal Bayes*

If the Bayesian has a limited memory and retrieval capacity, or if she has a generalization task, where new, unknown profiles may appear, she is forced to simplify the fully connected network. Figure 6 shows a rudimentary Bayesian network which neglects all interdependencies between cues, the so-called Naïve Bayes (or Idiot Bayes). If Naïve Bayes has no specific prior distribution but simply assumes that the chances of an object scoring higher than an other are uniformly distributed (i.e., $p(X_A > X_B) = p(X_B > X_A)$) then the posterior odds are

$$\frac{p(X_A>X_B|A\equiv(100101010),B\equiv(011000011))}{p(X_B>X_A|A\equiv(100101010),B\equiv(011000011))} =$$

$$\frac{p(C_1(A)=1,C_1(B)=0)|X_A>X_B)\times\cdots\times p(C_9(A)=0,C_9(B)=1)|X_A>X_B)}{p(C_1(A)=1,C_1(B)=0)|X_B>X_A)\times\cdots\times p(C_9(A)=0,C_9(B)=1)|X_B>X_A)} =$$

$$\frac{p(X_A>X_B|C_1(A)=1,C_1(B)=0)\times\cdots\times p(X_A>X_B|C_9(A)=0,C_9(B)=1)}{p(X_B>X_A|C_1(A)=1,C_1(B)=0)\times\cdots\times p(X_B>X_A|C_9(A)=0,C_9(B)=1)} \qquad (8)$$

Both the first and last equality are a consequence of Bayes's Theorem in its version for likelihood ratios and the fact that $p(X_A > X_B) = p(X_B > X_A)$. This means that the posterior odds for Naïve Bayes are a simple expression in terms of factors, each of which is either the validity of a cue (as in the case of $p(X_A > X_B|C_1(A) = 1, C_1(B) = 0) = v_1$), or 1 minus the validity of a cue (as in the case of $p(X_A < X_B|C_1(A) = 1, C_1(B) = 0) = 1 - v_1$). When both cue values coincide on the objects of a pair, the corresponding factors are 0.5 both in the numerator and in the denominator and can, therefore, be omitted. The posterior odds for the pair (A, B) in the German city environment are given by

$$\frac{v_1 \times (1 - v_2) \times (1 - v_3) \times v_4 \times v_6 \times (1 - v_9)}{(1 - v_1) \times v_2 \times v_3 \times (1 - v_4) \times (1 - v_6) \times v_9}$$

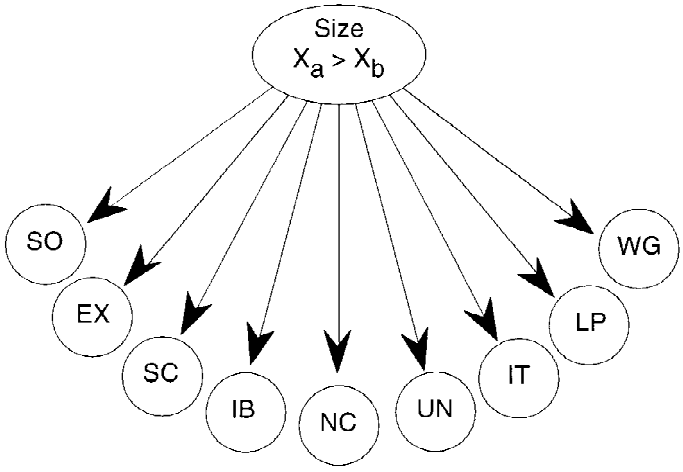

Figure 6. A simple Bayesian network for the German city data that assumes independence between the cues: Naïve Bayes. The nine cues are: 'Is the city the national capital?' (NC); 'Is the city a state capital?' (SC); 'Does the city have a soccer team in the major national league?' (SO); 'Was the city once an exposition site?' (EX); 'Is the city on the intercity train line?' (IT); 'Is the abbreviation of the city on license plates only one letter long?' (LP); 'Is the city home to a university?' (UN); 'Is the city in the industrial belt?' (IB); 'Is the city in former West Germany?' (WG). (Reprinted with permission of Oxford University Press.)

This is, all in all, a very simple formula for validities which can be automatically determined by checking corresponding cue values in the profiles. Forced to a deterministic answer Naïve Bayes will predict that A scores higher than B on the criterion, if the odds are larger than 1. Due to its simplicity, Naïve Bayes is the simple and fast—if not frugal— strategy used by Bayesians in a hurry.

Naïve Bayes corresponds to Take The Best and Franklin's Rule in that its performance is based solely on the cue validities. An even simpler Bayesian model which bets that all cue validities are equal (to some v with $0.5 < r < 1$) is what we call Minimal Bayes. Minimal Bayes will, thus, always make the same inference as Dawes's Rule.

5.3. *Friedman's network*

Assuming no dependencies between cues given criterion (Naïve Bayes) is a radical attitude. To assume that cues are all interdependent (Profile Memorization) is at least as radical. The Bayesian would like to strike a balance. She would like to detect those links between

cues that are relevant and robust. In other words, she would like to find a good Bayesian network for cue interdependencies.

In a Bayesian network the nodes with arrows pointing to a fixed node are called the *parents* of that node. The node itself is called a *child* of its parents. What follows is a fundamental rule for operating with Bayesian networks.

THEOREM 6 (The Markov Blanket). *The conditional probability of a node j existing in a certain state, given knowledge on the state of all other nodes in the network, is proportional to the product of the conditional probability of the node, given its parents multiplied by the conditional probability of each one of its children given its parents.*

In symbols:

$$\begin{aligned} &p(\text{node } j|\text{all other nodes}) = \\ &\quad = K \times p(\text{node } j|\text{parents of } j) \\ &\qquad \times \prod p(\text{child } k \text{ of } j \mid \text{parents of } k) \end{aligned} \tag{9}$$

where K is the normalizing constant (which cancels out in the posterior odds ratio, see (8)).

The set consisting of a node, its parents, its children and the other parents of its children is called the Markov Blanket of that node. What Theorem 6 states is that the Markov Blanket of a node determines the state of the node regardless of the state of all other nodes not in the Blanket (Pearl, 1988). The theorem, based essentially on Bayes's Rule, represents an enormous computational reduction for the storage and computations of probability distributions. It is precisely due to this type of reduction of computational complexity that Bayesian networks have become a popular tool both in statistics and in artificial intelligence in the last decade. Given a specific inference task and a set of data, the decision maker will search for a network which fits the data without fitting noise in the data, and based on that network she will make inferences.

A more accurate Bayesian network (than the one in Figure 6) needs to take into account the conditional dependencies between cues and the dependencies from hypothesis to cues. Of course, some dependencies are more relevant than others. Some may be so weak

that we may choose to ignore them to avoid overfitting. Thus, the Bayesian needs a Bayesian strategy for deciding which are the relevant links that should remain and which are the irrelevant ones that will only produce overfitting that should be pruned. She should find a feasible way to search through the possible networks and evaluate each network in terms of its performance. This search is infeasible without some heuristic which reduces the search space (as has been shown by Cooper, 1990, the general problem of Bayesian inference is NP-hard). One heuristic which is justified by Bayesian theory, and can be thought of as an instantiation of Occam's Razor, is to score Bayesian networks based on their posterior probability given the training set. If the posterior probability cannot be computed analytically, various estimators are available, such as the Bayes Information Criterion (Kass and Raftery, 1995). A number of algorithms have been developed that combine heuristic search over network structures with a scoring based on actual or approximate posterior probabilities. These algorithms find the Bayesian network which is among the most probable given the training set. The one we used here was put forward by Nir Friedman (e.g., Friedman and Goldszmit, 1996).

Figure 7 illustrates the Markov Blanket of the node Size, which represents the hypothesis: 'City *A* has more inhabitants than city *B*'. Obviously this node can be in two states (the other state is: 'City *B* has more inhabitants than city *A*'). According to Theorem 6,

$$\begin{aligned} p(\text{Size}|\text{UN,NC,IB,SO,EX,SC,IT}) &= \\ &= K \times p(\text{Size}|\text{SO,EX,SC}) \times p(\text{IB}|\text{Size,UN,NC}) \\ &\quad \times p(\text{IT}|\text{SO,EX,Size}) \end{aligned} \tag{10}$$

where K is the normalizing constant.

The number of probabilities to be estimated is still exponential in the number of parents per node, since each node stores a probability distribution for each combination of values for its parents. Again, the complexity of the problem may be further reduced by making structural assumptions constraining the probabilities. The algorithm we applied uses a simple classification tree to estimate the local probability tables. The classification tree greatly reduces the number of computations. Here, the problem of finding a good tree

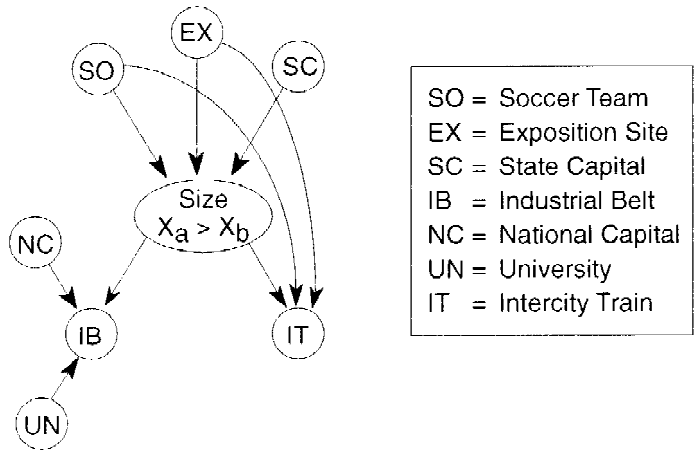

Figure 7. A savvy Bayesian Network. The Markov Blanket of Size (obtained by Friedman's search method) for predicting city population includes all nodes with the exception of LP and WG, which become irrelevant to Size, given the other cues. (Reprinted with permission of Oxford University Press.)

was solved with the same type of approach used to determine the network describing dependencies between cues, described above.

Figure 8 illustrates the tree that was produced by the program for p(Size | SO, EX, SC). The probability distribution for the Size variable is obtained by tracing the arcs of this tree. From Figure 8 we see that the first step is to check the exposition (EX) cue. If neither city is an exposition site, the probability is determined by whether the city has a soccer team (SO), and the state capital (SC) cue is irrelevant. Conversely, when one or both cities are exposition sites, then the probability distribution is determined by the state capital (SC) cue, and the soccer team (SO) cue is irrelevant. Thus, instead of requiring $2^7 = 128$ probabilities for the Size node given EX, SO, and SC, the tree representation of Figure 8 requires only $2^4 = 16$ probabilities. This is an important reduction of complexity.

To summarize, the method we used to find a probability model for the relationship between cue and criterion involves: (a) searching over a large space of directed graphs representing Bayesian networks on cue and criterion variables; (b) searching over decision tree models for quick estimation of local probabilities factoring the distribution of criterion given cues; (c) estimating the local probabilities; (d) computing the posterior distribution of the criterion variable given cue values using Equation (9). This method is quite

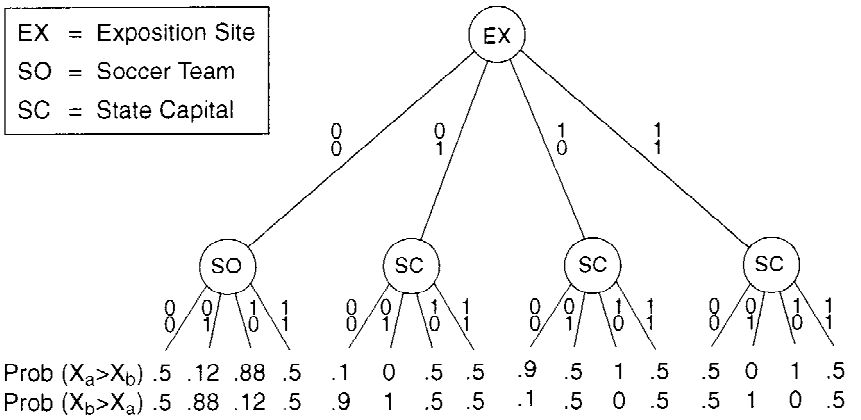

Figure 8. Tree for computing the local probability table for the Size node. If neither of the two cities A and B is an exposition site (symbolized by the two zeros in the left branch), then the only relevant cue is SO, that is, whether a city has a soccer team in the major league (SC is irrelevant). If A has a soccer team but B does not ('1' for A and '0' for B) then $p(X_A > X_B|\text{SO, EX, SC}) = 0.12$. (Reprinted with permission of Oxford University Press.)

complex. Does this complexity pay off when the resulting performance is compared to that of simple heuristics such as Take The Best?

5.4. *Performance of Bayesian networks*

To evaluate the performance of Take The Best, we compared it with that of Profile Memorization, Naïve Bayes, and the savvier Bayesian network described above, both in the fitting situation and under cross-validation. The size of the randomly chosen training set for cross-validation was 50% and we had 10 000 runs. Figure 9 illustrates the results. The Bayesian network found by means of Friedman and Goldszmit's method was handicapped by the small size of some of the data sets. In larger data sets (1000 objects or more) this type of Bayesian network becomes more robust (Friedman and Goldszmit, 1996). Both Take The Best and Naïve Bayes are robust on small sample sizes.

6. DISCUSSION

In this article we evaluated the fitness of simple models in three im-

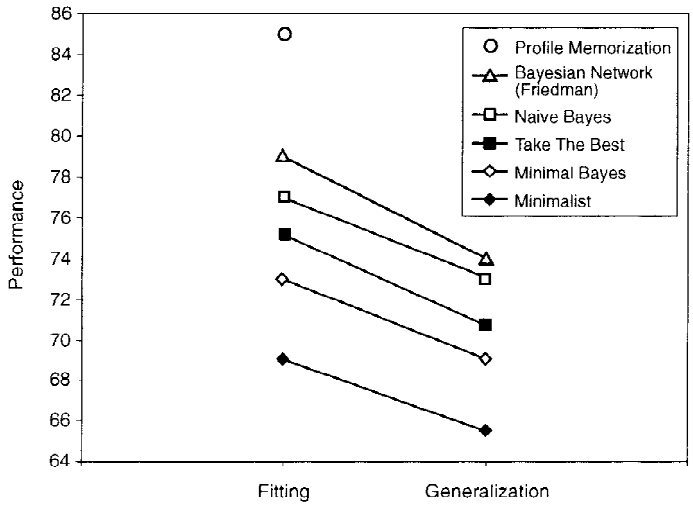

Figure 9. Performance of Take The Best, Minimalist, and four Bayesian Networks (Profile Memorization, the Bayesian Network à la Friedman, Naïve Bayes, and Minimal Bayes), averaged across the 20 different data sets described in Czerlinski et al. (1999). When cross validating the strategies, the size of the randomly chosen training set was 50%, and performance was averaged across 1000 runs.

portant classes: lexicographic, linear, and Bayesian models. Within each class we distinguished three subclasses characterized by their computational complexity in the preprocessing phase: minimal, simple, and sophisticated. We also identified the optimal model in each class, where optimality was defined with respect to performance when fitting known data. We demonstrated that simple models are fit, in particular, when generalizing to new data. A simple heuristic, which belongs to the class of lexicographic models, is Take The Best. It is simple because (a) its procedure for establishing the cue hierarchy is simple and (b) it is easily applied when comparing two objects. In the fitting case, Take The Best scored only a few percentage points below other lexicographic strategies that used more complex procedures to establish this hierarchy and achieved an even higher performance when generalizing to new data. Likewise, Franklin's Rule, a simple linear model, performed only one percentage point worse than multiple regression and outperformed it in the generalization task. Finally, Naïve Bayes, a simple Bayesian network, was superior to a savvier network (Friedman and Goldszmit, 1996), but only by two and one percentage points in the fitting and generalization case, respectively.

How large are these differences in performance? From a broader perspective, that is, on the full 100% scale, they appear insignificant (although there may be situations in which two percentage points are very important). Even if we consider the range between the worst and the best performance obtained in our tests (65% for Minimalist in the generalization task and 85% for Profile Memorization in the fitting task), it does not appear to make much difference whether one uses a simple or a complex model. This result is reminiscent of the phenomenon of flat maxima. If many sets of weights, even equal weights, can perform approximately as well as the optimal set of weights in a linear model, then it is called a flat maximum (Dawes and Corrigan, 1974; Dawes, 1979). The performance of simple models in some of the environments indicates that a flat maximum can extend beyond the issue of weights: Inferences based solely on the best cue can be approximately as accurate as those based on any weighted linear combination of all cues. The result in Section 4.3 (i.e., Theorem 3 on noncompensatory information), explains conditions under which we can expect flat maxima.

The scope of this work was analytical and theoretical rather than empirical. Yet, there has been an impressive amount of experimental studies focusing on the question whether people actually use fast and frugal heuristics. What distinguishes Take The Best from the other simple models is its frugality; Czerlinski et al. (1999) reported that Take The Best used 2.4 cues, whereas both Franklin's Rule and Naïve Bayes used 7.7 cues, averaged across 20 data sets. Therefore, Take The Best should be the strategy of choice when information is costly. In fact, Bröder (2000, Experiments 3 and 4) showed that in this situation more than 60% of participants were classified as using Take The Best, whereas none was classified as using Dawes's Rule. Payne et al. (1988, 1993) conducted several studies on how people choose their strategies according to the conditions imposed on them. For instance, if time is short, people tend to make inferences consistent with lexicographic models. Rieskamp and Hoffrage (1999) have provided further support: If the time available for making choices is progressively reduced, the number of choices that can be predicted by lexicographic models increases. These findings provide evidence for the hypothesis that people tend to use lexicographic strategies when search is costly and time is short. People are also sensitive

to noncompensatory information: In environments offering a cue, which is much more valid than all others, more decisions are made with a lexicographic rule (Martignon and Krauss, in press; Payne et al., 1988, 1993). Moreover, simple heuristics have also successfully been used as a basis for developing theories on the overconfidence phenomenon (Gigerenzer et al., 1991) or the hindsight bias (Hoffrage et al., 2000). These studies provided empirical evidence for hypotheses generated on the assumption that people use Take The Best.

In the present paper we focused on simple models for paired comparison tasks. The simplest heuristic for such a task is the recognition heuristic, which in the case of missing knowledge, is a building block of Take The Best. It can be applied if one object is recognized and the other not—in such a situation the recognition heuristic chooses the recognized object (for analytical results and empirical evidence that people use this heuristic see Goldstein and Gigerenzer, 1999; for more empirical evidence see also Ayton and Önkal, 1997; for the success of this heuristic on the stock market see Borges et al., 1999). There are other fast and frugal heuristics, essentially lexicographic, which have been designed for solving categorization tasks (i.e., Berretty et al., 1999, proposed the Categorization By Elimination heuristic, and Dhami and Harris, 2001, proposed the matching heuristic; for empirical evidence supporting the hypothesis that people use these heuristics, see Berretty, 2001, and Dhami and Harris, 2001). Two recent publications by Gigerenzer and Selten (2001), and Todd, Gigerenzer, and the ABC Research Group (2000), provide overviews and discussions of various heuristics and models of bounded rationality including sections on their appropriateness as behavioral models.

The results reported in this paper were obtained with real-world data but must be evaluated with respect to the conditions used, which include the following. First, we studied inferences only under complete knowledge, unlike Gigerenzer and Goldstein (1996), who studied the performance of heuristics also under limited knowledge. Limited knowledge (e.g., knowing only a fraction of all cue values) is a realistic condition that applies to many situations in which predictions must be made. In the simulations reported by Gigerenzer and Goldstein, the major result was that the more limited the know-

ledge is, the smaller the discrepancy becomes between Minimalist and other algorithms. Thus, Minimalist, whose respectable scores were nevertheless always the lowest, really flourishes when there is only limited knowledge. Other conditions of the studies reported here include the use of binary and dichotomized data, which can be a disadvantage to multiple regression and Bayesian networks. Finally, we have used only correct data and have not studied predictions under the realistic assumption that some of the information is incorrect.

Another direction along which the present work should be extended is the study of information structure in the environment (Martignon and Hoffrage, 1999). Such a program is a Brunswikian program, but it is one that dispenses with multiple regression as the tool for describing both the processes of the mind and the structure of the environment. Fast and frugal algorithms can be ecologically rational in the sense that they exploit specific and possibly recurrent characteristics of the environment's structure. Models of reasonable judgment should look outside the mind, to its environment. As Gigerenzer and Goldstein (1996) stated: 'Models of reasonableness do not have to forsake accuracy for simplicity. The mind can have it both ways'.

ACKNOWLEDGEMENTS

Each of the authors made large substantial contributions to this article. We thank Nathan Berg, Valerie Chase, Gerd Gigerenzer, Elke Kurz, Peter Todd and two anonymous reviewers for helpful comments on previous drafts, several members of the ABC research group for useful discussions, and Niko Kriegeskorte, Torsten Mohrbach, and Valentin Zacharias for programming the simulations. We also thank the German Research Foundation (Grant Ho 1847/1) for financial support.

APPENDIX

Proof of Eq. 2. We have to show that the following equation holds:

$$\frac{R}{R+W} = \frac{S_0 - \dfrac{N_0(N_0+1)}{2}}{N_0 N_1} \qquad (*)$$

where R and W denote the number of right and wrong inferences, respectively, in the complete paired comparison task, N_0 and N_1 are the numbers of 0 and 1 entries in the matrix that builds the environment, and S_0 denotes the sum of all ranks of 0 entries (the ranks are given to the objects ordered according to the criterion; the object with the highest criterion value has rank 1).

Let us begin by considering a cue with validity 1. There is a number K, satisfying $1 \leq K < N$, such that the K objects with the largest criterion values have a value of 1 on the cue, while the remaining $(N - K)$ objects have a value of 0. Thus $N_0 = (N - K)$ and $N_1 = K$ (see Cue 1 in the following table).

Rank	Object	Cue 1	Cue 2
1	O_1	1	1
2	O_2	1	1
..	..	1	1
K	O_K	1	0
$K+1$	O_{K+1}	0	1
$K+2$	O_{K+2}	0	0
..	..	..	..
$N-1$	O_{N-1}	0	0
N	O_N	0	0

To compute S_0 we make use of the formula for the sum of an arithmetic progression, obtaining

$$S_0 = \frac{N(N+1) - K(K+1)}{2}$$

Replacing R (which for a cue with validity of 1 amounts to $K(N-K)$) and S_0 in $(*)$ and canceling the numerators of the two expressions in $(*)$ (which can be done because $R + W$, the total number of

inferences performed by the cue, is equal to N_0N_1) yields

$$K(N-K) = \frac{N(N+1) - (K(K+1)) - (N-K)(N-K+1)}{2}$$

which is easily obtained by multiplying out the terms on the right side and performing the necessary cancellations. Thus, we have shown that $(*)$ holds for a cue with validity 1.

The next step is to perform exactly one inversion in the cue values of Cue 1: For the two objects O_K and O_{K+1} we change the 1 into a 0 and the 0 into a 1, thereby obtaining Cue 2 (see Table). The number of 0s and the number of 1s for this cue are the same for Cue 1, and thus $R + W$ as well as N_0N_1 are the same for both cues. However, Cue 2 makes exactly one less correct inference than Cue 1. Clearly for Cue 2 S_0 is also smaller than S_0 for Cue 1, and the difference is 1. Thus, the nominators of (1) and (2) (i.e., R and S_0, respectively) have both been diminished by 1, while their denominators remain the same. This proves our assertion for the cue obtained by performing one inversion in a cue of validity 1. Because any cue can be obtained by performing successive inversions on a cue of a validity 1, and because none of these inversions will change the identity between (1) and (2), as can be shown by complete induction, our assertion is true in general. □

Proof of Theorem 3. We now prove that Take The Best is equivalent—in performance—to a linear model with noncompensatory weights, where the highest weight corresponds to the cue with highest validity, the second highest weight to the cue with second highest validity, and so on. Consider an environment provided with a set of cues $Q = \{q_1, q_2, \ldots, q_M\}$, where $V(q_1) \geq V(q_2) \geq \cdots \geq V(q_M)$, and $V(q_i)$ is the validity of q_i. Define the score $s(O_j)$ of an object O_j by

$$s(O_j) = q_1(O_j)\frac{1}{2} + q_2(O_j)\frac{1}{2^2} + q_3(O_j)\frac{1}{2^3} + \cdots + q_M(O_j)\frac{1}{2^M}$$

We have to show that the lexicographic comparison of objects O_i and O_j implies that $O_i > O_j$ if and only if $s(O_i) > s(Oj)$.

Proof. Assume that the best cue q_1 in Q satisfies $q_1(O_i) > q_1(O_j)$. Because we only consider binary cues, this assumption implies that $q_1(O_i) = 1$ and $q_1(O_j) = 0$.

Since

$$\frac{1}{2} > \frac{1}{2^2} + \frac{1}{2^3} + \frac{1}{2^4} + \cdots + \frac{1}{2^M}$$

the score $s(O_i)$ is certainly larger than the score $s(O_j)$. For the reciprocal observe that it is impossible to have

$$\frac{1}{2}q_1(O_i) + \frac{1}{2^2}q_2(O_i) + \cdots + \frac{1}{2^M}q_M(O_i) > \frac{1}{2}q_1(O_j) + \frac{1}{2^2}q(O_j) + \cdots + \frac{1}{2^M}q(O_j)$$

unless $q_1(O_i) = 1$ and $q_1(O_j) = 0$, based on the same argument that 1/2 is larger than any sum of its lower powers. The same is valid if the first differing cue value occurs at some other column of the cue matrix. □

Proof of Theorem 4. Consider a reference class R of N objects $(O_1, \ldots, O_i, \ldots O_n)$. We define the score $s(O_i)$ of O_i as the sum of entries in its cue profile $Q(O_i)$. Theorem 4 holds if, in the cue matrix containing all favorable cues but not the secure cues, there are more ones in each row than in the following row (objects are rank ordered according to the criterion, with the object scoring highest in the first row). Consider the i-th row and its successor the $(i + 1)$-th row. Cues contained in the matrix that have the same values in the i-th and the $(i + 1)$-th entries are irrelevant to the theorem because they add equally to both sums.

The set of all cues in the matrix that have a 1 in the i-th entry and a 0 in the $(i + 1)$-th entry will be called M_i. The set of all cues in the matrix that have a 0 in the i-th entry and a 1 in the $(i + 1)$-th entry will be called M_{i+1}. It it sufficient to prove that the set M_i is larger than the set M_{i+1}.

For each element of M_{i+1} changing the 0 in the i-th entry into a 1 and the 1 in the $(i + 1)$-th entry into a 0 yields a cue of higher validity, because one inversion is transformed into an agreement. The resulting cue will be in the matrix (and thus an element of the set M_i) except when its validity becomes 1 after the transformation.

This can only happen for the cue which has all ones above the i-th entry and all zeros below the $(i + 1)$-th entry. It has thus been shown, that the set M_i is at least the size of the set M_{i+1} minus 1: inverting the i-th and $(i + 1)$-th entries maps all elements of M_{i+1} onto elements of M_i except for one element of M_{i+1}, which is mapped onto a cue of validity 1.

If there are two elements of M_i that are not elements of M_{i+1} with their i-th and $(i + 1)$-th entries inverted, then the set M_i is larger than the set M_{i+1}, and the theorem is proven. We will show that it is possible to construct two cues that:

- have a 0 in entry i and a 1 in entry $i + 1$
- are not in M_{i+1} because their validities are equal to or below 0.5
- can be transformed to elements of M_i by inverting their i-th and the $(i + 1)$-th entries (The inversion increases their validity so that it falls within]0.5, 1[.)

The two cues resulting from the inversion are therefore in the set M_i without being elements of M_{i+1} with their i-th and $(i + 1)$-th entries inverted. The two cues are therefore not within the subset of cues of M_i accounted for in the previous step. If the existence of two such cues can be shown then the set M_i is larger than the set M_{i+1} and the theorem is proven.

Case A: N is odd

If N odd, then there is one central entry to the cue. In that case there are at least two ways to construct a symmetric cue that has a 0 in the i-th and a 1 in the $(i+1)$-th entry: Entries i and $i+1$ will be mirrorred vertically around the cue's central entry. This will determine three or four entries of the cue: Three if the i-th or the $(i + 1)$-th entry is the cue's central entry and four if it is not. The number of objects N is defined to be larger than 4, so there is at least one entry that can be set to 1 or 0 to obtain two symmetrical cue.

Since the obtained cues are symmetrical their validity is 0.5. Inverting the i-th and $(i + 1)$-th entries will increase their validities so that they will fall within]0.5, 1[and be elements of M_i. □

Case B: N is even

If N is even and $i \neq N/2$, two symmetrical cues that have a 0 in the i-th and a 1 in the $(i + 1)$-th entry can be constructed as explained above: by mirroring the i-th and $(i + 1)$-th entries around the cue's central axis and filling the remaining entries in two different ways to obtain two symmetrical cues. Inverting the i-th and $(i + 1)$-th entries, again, will increase the validities so that they will fall within]0.5, 1[and be elements of M_i.

If N is even and $i = N/2$, a slightly different approach must be taken: Whereas the i-th entry is set to 0 and the $(i + 1)$-th to 1 as before, the other entries are either all set to 1 or they are all set to zero. The resulting cues are symmetrical except for one central inversion. Their validities are thus below 0.5. Inverting the i-th and $(i+1)$-th entries, will yield cues that are symmetrical except for one central agreement. Their validities will thus fall within]0.5, 1[and they will be elements of M_i. □

REFERENCES

Ayton, P. and Önkal, D. (1997), Forecasting football fixtures: Confidence and judged proportion correct, Unpublished manuscript.

Berretty, P. M. (2001), Cue preference in a multidimensional categorizataion task, Manuscript submitted for publication.

Berretty, P. M., Todd, P. M. and Martignon, L. (1999), Using few cues to choose: Fast and frugal categorization, In G. Gigerenzer, P. M. Todd and the ABC Research Group, *Simple Heuristics That Make Us Smart* (pp. 235–254), New York: Oxford University Press.

Borges, B., Goldstein, D. G., Ortmann, A. and Gigerenzer, G. (1999), Can ignorance beat the stock market?, in G. Gigerenzer, P. M. Todd and the ABC Research Group, *Simple Heuristics That Make Us Smart* (pp. 59–72), New York: Oxford University Press.

Breiman, L., Friedman, J. H., Olshen, R. A. and Stone, C. J. (1993), *Classification and Regression Trees*, New York: Chapman and Hall.

Bröder, A. (2000), Assessing the empirical validity of the 'Take The Best' heuristic as a model of human probabilistic inference, *Journal of Experimental Psychology: Learning, Memory, and Cognition* 26: 1332–1346.

Cooksey, R. W. (1996), *Judgment Analysis: Theory, Methods, and Applications*, San Diego, CA: Academic Press.

Cooper, G. (1990), The computational complexity of probabilistic inferences. *Artificial Intelligence* 42: 393–405.

Czerlinski, J., Gigerenzer, G. and Goldstein, D. G. (1999), How good are simple heuristics?, in G. Gigerenzer, P. M. Todd, and the ABC Research Group, *Simple Heuristics That Make Us Smart* (pp. 97–118), New York: Oxford University Press.

Dawes, R. M. (1979), The robust beauty of improper linear models in decision making, *American Psychologist* 34: 571–582.

Dawes, R. M. and Corrigan, B. (1974), Linear models in decision making, *Psychological Bulletin* 81: 95–106.

Dhami, M. and Harris, C. (2001), Fast and frugal versus regression models of human judgement, *Thinking and Reasoning* 7: 5–27.

Friedman, N. and Goldszmit, M. (1996), Learning Bayesian networks with local structure, in *Proceedings of the 12th Conference on Uncertainty in Artificial Intelligence (UAI)* (pp. 252–262), San Mateo, CA: Morgan Kaufmann.

Garey, M. R. and Johnson, D. S. (1979), *Computers and Intractability: A Guide to the Theory of NP-Completeness*, San Francisco, CA: W. H. Freeman.

Gigerenzer, G. (1981), *Messung und Modellbildung in der Psychologie*, Munich: Ernst Reinhard Verlag.

Gigerenzer, G., Czerlinski, J. and Martignon, L. (1999), How good are fast and frugal heuristics? in J. Shanteau, B. Mellers, and D. Schum (eds.), *Decision Research from Bayesian Approaches to Normative Systems: Reflections on the Contributions of Ward Edwards*. Norwell, MA: Kluwer Academic Publishers.

Gigerenzer, G. and Goldstein, D. G. (1996), Reasoning the fast and frugal way: Models of bounded rationality, *Psychological Review* 103: 650–669.

Gigerenzer, G. and Hoffrage, U. (1995), How to improve Bayesian reasoning without instruction: Frequency formats, *Psychological Review* 102: 684–704.

Gigerenzer, G., Hoffrage, U. and Kleinbölting, H. (1991). Probabilistic mental models: A brunswikian theory of confidence, *Psychological Review* 98: 506–528.

Gigerenzer, G. and Selten, R. (eds.) (2001), *Bounded Rationality: The Adaptive Toolbox*, Cambridge, MA: MIT Press.

Gigerenzer, G., Todd, P. M. and the ABC Research Group (1999), *Simple Heuristics That Make Us Smart*, New York: Oxford University Press.

Goldstein, D. G. and Gigerenzer, G. (1999), *How ignorance makes us smart: The recognition heuristic*, in G. Gigerenzer, P. M. Todd and the ABC Research Group, *Simple Heuristics That Make Us Smart* (pp. 37–58), New York: Oxford University Press.

Hasher, L. and Zacks, R. T. (1984), Automatic processing of fundamental information: The case of frequency of occurrence, *American Psychologist* 39: 1372–1388.

Hoffrage, U., Hertwig, R. and Gigerenzer, G. (2000), Hindsight bias: A by-product of knowledge updating?, *Journal of Experimental Psychology: Learning, Memory, and Cognition* 26: 566–581.

Hoffrage, U., Lindsey, S., Hertwig, R. and Gigerenzer, G. (2000), Communicating statistical information, *Science* 290: 2261–2262.

Holte, R. C. (1993), Very simple classification rules perform well on most commonly used datasets, *Machine Learning* 3(11): 63–91.

Kahneman, D., Slovic, P. and Tversky, A. (1982), *Judgment under Uncertainty: Heuristics and Biases*, New York: Cambridge University Press.

Kass, R. and Raftery, A. (1995), Bayes Factors, *Journal of the American Statistical Association* 90: 430.

Krauss, S., Martignon, L. and Hoffrage U. (1999), Simplifying Bayesian inference: The general case, in L. Magnani, N. Nersessian and P. Thagard, (eds.), *Model-Based Reasoning in Scientific Discovery* (pp. 165–179), New York: Plenum Press.

Kurz, E. and Martignon, L. (1999), Weighing, then summing: The triumph and tumbling of a modeling practice in psychology, in L. Magnani, N. Nersessian and P. Thagard, (eds.), *Model-Based Reasoning in Scientific Discovery* (pp. 26–31), Pavia: Cariplo.

Lages, M., Hoffrage, U. and Gigerenzer, G. (1999), How heuristics produce intransitivity and how intransitivity can discriminate between heuristics, Manuscript submitted for publication.

Martignon, L. and Hoffrage, U. (1999), Why does one-reason decision making work? A case study in ecological rationality, in G. Gigerenzer, P. M. Todd, and the ABC Research Group, *Simple Heuristics That Make Us Smart* (pp. 119–140), New York: Oxford University Press.

Martignon, L. and Krauss, S. (in press), Can l'homme éclairé be fast and frugal? Reconciling Bayesianism and bounded rationality, in S. Schneider and J. Shanteau (eds.), *Emerging Perspectives on Decision Research*, Oxford, UK: Oxford University Press.

Martignon, L. and Laskey, K. B. (1999), Bayesian benchmarks for fast and frugal heuristics, in G. Gigerenzer, P. M. Todd and the ABC Research Group, *Simple Heuristics That Make Us Smart* (pp. 169–188), New York: Oxford University Press.

Martignon, L. and Schmitt, M. (1999), Simplicity and robustness of fast and frugal heuristics, *Minds and machines* 9: 565–593.

Payne, J. W., Bettman, J. R. and Johnson, E. J. (1988), Adaptive strategy selection in decision making, *Journal of Experimental Psychology: Learning, Memory, & Cognition* 14: 534–552.

Payne, J. W., Bettman, J. R. and Johnson, E. J. (1993), *The Adaptive Decision Maker*, New York: Cambridge University Press.

Pearl, J. (1988), *Probabilistic Reasoning in Intelligent Systems*, San Francisco, CA: Morgan Kaufmann.

Rieskamp, J. and Hoffrage, U. (1999), When do people use simple heuristics, and how can we tell?, in G. Gigerenzer, P. M. Todd and the ABC Research Group, *Simple Heuristics That Make Us Smart* (pp. 141–167). New York: Oxford University Press.

Rivest, R. J. (1987), Learning decision lists, *Machine Learning* 2: 229–246.

Shannon, C. (1948), A mathematical theory of communication, *Bell Systems Technical Journal* 27: 379–423, 623–656.

Slegers, D. W., Brake, G. L. and Doherty, M. E. (2000), Probabilistic mental models with continuous predictors, *Organizational Behavior and Human Decision Processes* 81: 98–114.

Todd, P.M., Gigerenzer, G. and the ABC Research Group (2000), How can we open up the adaptive toolbox? (Reply to commentaries) *Behavioral and Brain Sciences* 23: 767–780.

Tversky, A. and Kahneman, D. (1974), Judgment under uncertainty: Heuristics and biases, *Science* 185: 1124–1131.

Address for correspondence: Laura Martignon, Center for Adaptive Behavior and Cognition, Max Planck Institute for Human Development, Lentzeallee 94, D-14195, Berlin, Germany. Phone: +49-30-82406-355; Fax: +49-30-82406-394; E-mail: martignon@mpib-berlin.mpg.de