"MUST HAVE" ASPECTS VS. TRADEOFF ASPECTS IN MODELS OF CUSTOMER DECISIONS

JOHN R. HAUSER
MIT
ELY DAHAN
UCLA
MICHAEL YEE
MIT LINCOLN LABORATORIES
JAMES ORLIN
MIT

This paper provides an applied managerial summary of the theory and empirical tests developed by the authors in a forthcoming *Marketing Science* article. For detailed derivations, proofs of all propositions, screen shots of the web-based data collection, details on the empirical analysis, detailed statistical tests, and other material please refer to:

Michael Yee, John Hauser, James Orlin, and Ely Dahan (2006), "Greedoid-Based Non-compensatory Two-Stage Consideration-then-Choice Inference," (April) forthcoming, Marketing Science.

"MUST-HAVE" FEATURES

There are over 300 make-model combinations of automobiles on the market, but the average consumer considers less than 10 make-model combinations. If an automobile manufacturer can determine how to entice consumers to consider its make-model combinations, e.g., a Ford Mustang, then the manufacturer can reduce the choice set from 1 in 300 to 1 in 10—a factor of 30. By designing cars that will be considered, an automobile manufacturer greatly increases its chances of making a sale. For example, General Motors (GM) believes that GM vehicles are better than consumers perceive them to be and that GM would gain share if consumers would be more willing to consider GM vehicles.

The consideration-set challenge is not limited to automobiles. A recent survey of websites selling personal digital assistants (PDAs) suggests that there are 21 models available at Circuit City, 25 at Staples, 27 at Microcenter, 30 at CompUSA, and 97 at Govconnection (MIT's approved vendor). It is the rare consumer who will evaluate all of the PDAs available before making a decision. More likely, the consumer will screen on some characteristics before evaluating a small subset of the available PDAs.

In each of these cases the managerial challenge is to identify the "must have" (or "must not have") features that determine the consideration sets of consumers. Must-have features are non-compensatory in the sense that a product with must-have features is preferred to a product without these must-have features even if the product without the must-have features is better on all other features.

The identification of must-have features is related to the conjoint-analysis goal of identifying the most important features, but there are differences. We seek to complement conjoint analysis methods. In particular, we address two issues: (1) We seek to infer directly non-compensatory decision processes, e.g., processes in which some products have must-have features that drive consideration or choice. (2) We explore methods that apply well if the respondent is asked to indicate those profiles which he or she would consider. This task can be used alone or in addition to the rank, rating, or choice tasks that are typical in conjoint analysis. The methods we summarize are practical. Empirical data suggests they often predict at least as well as traditional conjoint analysis.

CONJOINT ANALYSIS ASSUMES, PRIMARILY, A COMPENSATORY MODEL

Traditional conjoint analysis represents products by profiles of features and asks respondents to express their preferences among those profiles. Respondents might rank the profiles, provide ratings that express preference, or simply choose from sets of profiles. The basic preference model assumes that the preference for a profile can be expressed as a combination of the "partworths" of the feature levels. In most cases, the preference function is separable, meaning that a preference score is an additive or multiplicative combination of the partworths for the levels of the features that describe the profile. Although interaction terms are possible, they are not commonly used. For ease of exposition we ignore interactions, although this does not affect our basic arguments.

Figure 1 presents an additive model that illustrates how a respondent might evaluate two smartphones using a compensatory decision process. In Figure 1, the partworth for "Verizon" is larger than the partworth for "Cingular," but the other partworths of the Cingular phone, "flip," "price = $299," and "manufactured by Sony," compensate for the fact that, all else equal, the respondent would prefer "Verizon" to "Cingular." An additive conjoint model predicts that the respondent would choose the Cingular smartphone.

Figure 1
Illustration of a Compensatory Model

"Utility = sum of "partworths"

Verizon = 10	Cingular = 3
Brick = 2	Flip = 7
$499 = 1	$299 = 6
…	…
Nokia = 3	Sony = 5

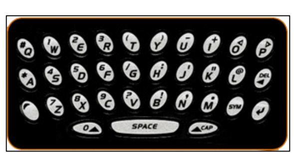
"Flip" and "$299" can compensate for a lower partworth on "carrier."

But what if the respondent is faced with over twenty smartphones and adopts a different, more-realistic screening rule. As indicated in Figure 2, the respondent might consider only flip phones, with mini-keyboards, from Blackberry. If this were the case, smartphones without these characteristics would never be considered and would never be chosen. In a non-compensatory decision process, other features such as carrier, operating system, GPS capability, camera capability, and price cannot compensate for a smartphone that does not flip, have a mini-keyboard, or is from Blackberry.

Figure 2
Illustration of a Non-Compensatory Decision Process

"I will only consider flip phones, with mini-keyboards, from Blackberry"

Flip



Mini Keyboard

BLACKBERRY

Phone Brand

In theory, an additive partworth model can represent such a process, but, in practice, such an additive model would put a strain on estimation. We illustrate this capability with binary features for ease of exposition. For binary features it is easy to identify a set of partworths that acts lexicographically. Lexicography is a non-compensatory process in which the respondent evaluates profiles by features, one at a time, accepting that profile only if it is better or equal to other profiles on the feature currently being evaluated. The respondent continues until all profiles have been sorted according to the task—full rank if traditional full-profile conjoint analysis, best of a set for choice-based conjoint analysis, or "considered" for a consideration task.

One function that acts lexicographically for n binary features is to assign 2^{n-1} to the first evaluated feature, 2^{n-2} to the next feature, …, and 1 to the last evaluated feature. For sixteen binary features, this means that the partworth for the first feature would be 32,768, the second feature 16,384, and the last feature 1. Clearly, partworths that vary in magnitude by a factor of over thirty-two thousand would put a strain on most estimation procedures and would be extremely sensitive to response errors. Furthermore, mimicking a lexicographic process by assigning values to partworths does not imply unique partworths, even to a positive linear transformation. Other combinations such as 3^{n-1}, 3^{n-2}, …, 1 would also work. In fact, still another set of "partworths," the denominations of US currency, acts lexicographically. A rational consumer would prefer 5¢ to 1¢, 10¢ to 5¢ plus 1¢, 25¢ to 10¢ plus 5¢ plus 1¢, etc. for 50¢, $1, $2, $5, $10, $20, $50, $100, all the way up to $10,000 – a form of currency no longer in circulation.

CONSIDERATION TASK

The standard task in conjoint analysis asks the respondent to indicate preferences. Although most estimation procedures could be modified to deal with data in which the respondent simply expresses consideration, that has not been common. While we expect that non-compensatory processes are common for choice tasks, we expect they are even more common for consideration tasks. Thus, we want a method that handles consideration-set data naturally, as well as choice, ranking, or rating data.

NON-COMPENSATORY MODELS

In this paper we consider lexicographic models, an important class of non-compensatory models. As defined above for binary features, a respondent acts lexicographically if he or she first ranks the <u>features</u> of profiles. Those profiles with the first-ranked feature are preferred to those profiles without the first ranked feature. In the case of ties on the first-ranked feature, we move to the next profile continuing until the tie is broken. A profile with a higher-ranked feature is chosen even if another profile without the high-ranked feature is better on each and every lower-ranked feature.

For example, consider four binary features of smartphones: flip preferred to brick, small preferred to large, mini-keyboard preferred to no keyboard, and a Palm operating system preferred to a Microsoft operating system. Suppose the <u>features</u> are ranked flip, small, mini-keyboard, and Palm. Then if we represent profiles by indicting the features they have, this lexicographic ordering implies the following orders. We have underlined the critical comparison.

- {flip, large, no keyboard, Microsoft} $\succ$ {brick, small, keyboard, Palm}
- {flip, small, keyboard, Palm} $\succ$ {flip, large, keyboard, Palm}
- {brick, large, keyboard, Microsoft} $\succ$ {brick, large, no keyboard, Palm}
- {brick, large, keyboard, Palm} $\succ$ {brick, large, keyboard, Microsoft}.

The basic idea extends to multi-level features, but, in order to model consumer decision making, we must consider how consumers treat levels within features. Here we adopt Tversky's (1972) nomenclature and call each level of a feature an "aspect." Basically, an aspect is a binary feature, e.g., "flip vs. brick" or "Verizon vs. not Verizon." A multi-level feature is then a set of linked aspects. For example, the feature, "carrier," can have one of four aspects: Verizon, Cingular, Sprint, or Nextel.

For multi-level features we can have four different decision processes that vary on how consumers evaluate aspects (levels) within features. These are:

- lexicographic by features (LBF): the consumer first ranks the features and then ranks aspects within features
- acceptance by aspects (ABA): the consumer first ranks aspects and then accepts profiles if they have the aspects
- elimination by aspects (EBA): the consumer first ranks aspects and then rejects profiles if they do not have that aspect
- lexicographic by aspects (LBA): the consumer first ranks aspects and then either accepts a profile if it has an aspect or rejects a profile if it does not have an aspect

These four decision rules are illustrated in Figure 3 for playing cards. Note that (1) if all features are binary, then all four decision models can provide the same ordering of profiles, (2) if some features are multi-level, then the decision models provide different ordering of profiles, and (3) LBA nests all of the other decision models.[1]

[1] When we say that two decision models can provide the same ordering of profiles, we mean, there exists an aspect- or feature-ordering that orders the profiles the same. The aspect- or feature-ordering can be different in the two models being compared. For example, for a given ABA model there exists an EBA model that ranks profiles in the same order. To transform an ABA model into an EBA model, we reverse all binary aspects (e.g., change Verizon to not Verizon) and reverse the ranking of aspects. We can always represent an LBF model by an ABA model if we constrain aspects within a feature to be contiguous in a rank order, but we can identify ABA models that are not represented by equivalent LBF models. We can represent any LBF, ABA, or EBA model by an equivalent LBA model.

Figure 3
Four Lexicographic Heuristics Illustrated with Playing Cards

Simplifying Heuristic	TLA (*Three-Letter Acronym*)	Ranking Rule	First Choice	by 2nd Choice	3rd Choice	Totally Diverge by 4th Choice	5th Choice	6th Choice	7th Choice	Last Choice
Lexicographic By Features	LBF	(♠ > ♥ > ♦ > ♣), (A > J)	A♠	J♠	A♥	J♥	A♦	J♦	A♣	J♣
Acceptance By Aspects	ABA	♠, A, ♥, ♦	A♠	J♠	A♥	A♦	A♣	J♥	J♦	J♣
Elimination By Aspects	EBA	~~♣~~, ~~J~~, ~~♦~~, ~~♥~~	A♠	A♥	A♦	J♠	J♥	J♦	A♣	J♣
Lexicographic By Aspects	LBA	~~♣~~, ♠, A, ♥	A♠	J♠	A♥	A♦	J♥	J♦	A♣	J♣

WHY NOT JUST ENUMERATE ALL LEXICOGRAPHIC MODELS?

A lexicographic model is a simple model and it is an easy decision rule for a consumer to use. Furthermore, while the set of all compensatory models is defined on an uncountable infinite set of partworth values ($\Re^{n-1}$ for n aspects), there are only finitely many lexicographic orderings of aspects. For example, if we know which aspects are favorable, e.g., the respondent prefers flip smartphones to brick smartphones, then there are $n!$ possible lexicographic orderings of the n aspects.[2] If we need to infer which aspects are favorable, then there are $2^n n!$ possible lexicographic orders.

If the number of aspects is small, then it is feasible to enumerate exhaustively all possible lexicographic orders and choose, for each respondent, the lexicographic order of <u>aspects</u> that best describes that respondent's profile ordering (rating or choice). For example, with 3 aspects we need only consider $3! = 6$ aspect orders for ABA or EBA and only $2^3 3! = 48$ aspect orders for LBA. With 4 aspects we need only consider $4! = 24$ and $2^4 4! = 384$ aspect orders for ABA and LBA, respectively.

With exhaustive enumeration we could use any reasonable metric to define "best." For example, we might maximize a Spearman rank correlation, a Kendall's τ rank correlation, or, perhaps, the number of paired violations.[3]

The challenge comes when the number of aspects grows. For example, Martignon and Hoffrage (2002) study a famous lexicographic problem and exhaustively enumerate all lexicographic orders for a 9-aspect problem. Their problem requires a total of $9! = 362{,}800$ orders to be evaluated. They report computations that required two days to complete. On

[2] If the model is constrained to LBF there are fewer possible orderings ($m_1!m_2!$ for m_1 features at $m_2!$ levels each).
[3] Minimizing the number of paired violations is equivalent to maximizing Kendall's τ.

today's computers we can do such computations much faster, perhaps minutes or even seconds. With 300 respondents and 9 aspects, exhaustive enumeration might still be feasible for 9 aspects. However, when we increase the challenge to a 16-aspect problem, then we must evaluate 16! aspect orders for ABA. However, 16!= 57,657,600 * 9! if we knew the preferred direction of each aspect. It is much worse if we must infer the preferred direction of each aspect: $2^{16}16!$ = 3,778,648,473,600 * 9!. If the Martignon and Hoffrage algorithm took just a single second to run, an LBA problem would take almost 120,000 years. Clearly, exhaustive enumeration of reasonably-sized problems will not be feasible any time soon, even taking Moore's Law of increased computing power into account.

Greedoid Methods Developed by Yee, et. al. (2006)

Fortunately, the relationship between an aspect order and an induced profile order has special structure. Yee, et. al. (2006) establish that this relationship can be described by a greedoid language if the goodness of fit measure is to minimize the number of paired comparisons that are violated. Based on the mathematics developed for greedoid languages, Yee, et. al. prove the following propositions.

1. if there exists a lexicographic ordering on the aspects that induces a profile ordering that matches the respondent's profile ordering, then the aspect ordering can be found with a polynomial-time algorithm (a greedy algorithm).
2. if there is no lexicographic ordering on the aspects that induces a profile ordering that perfectly matches the respondent's profile ordering, then the best-fitting aspect ordering (or orderings) can be found with a dynamic program that runs in the order of 2^n steps.
3. the dynamic programming algorithm can be altered slightly to allow the paired orderings to be weighted differentially. A weighted algorithm also runs in the order of 2^n steps.

Yee, et. al. and, later, Yee (2006) prove, in addition, a series of propositions that speed the algorithm further. The net result is that there is a simple algorithm that can solve reasonably-sized problems in a second or less. (The simple algorithm requires only a dozen lines of pseudo-code. More complicated algorithms require more code, but run significantly faster.) Yee, et. al. further demonstrate that if the manager is only interested in the aspects that are ranked highly, then even large problems, say 50 aspects, can be solved in reasonable time.

With the greedoid-based dynamic programming algorithms it is now feasible to use realistic preference, choice, or consideration tasks to infer the best lexicographic ordering of the aspects.

Benchmarks

We consider two traditional benchmarks. The first is hierarchical Bayes and the second LINMAP. In our empirical test, we collect data with either a rank-order data or a consider-then-rank task, thus the appropriate hierarchical Bayes model is a ranked-logit model, which we label HBRL. LINMAP is a traditional model based on linear programming (Srinivasan and Shocker 1973). However, recently, Srinivasan (1998) recognized that the performance of LINMAP could be improved by requiring that partworths be chosen to assure strict rank orderings of the profiles. In previous versions, if the respondent ranked profile *i* above profile *j*, then LINMAP's goodness-of-fit measure would not penalize a set of partworths that rated profile *i* equal to profile j. The new version of LINMAP adds a penalty for partworths that allow ties. (See Srinivasan 1998 for details.) We provide comparisons between the two versions of LINMAP.

There is an important complication that we must take into account when using either HBRL or LINMAP as benchmarks. HBRL and LINMAP are additive models and, hence, have the capability to identify partworths that act as if the respondent were using a lexicographic model. (Review Section 2.) Although finding such partworths may not be practical when there is response error, such partworths are theoretically feasible. Any benchmark comparisons must take this (theoretical) nesting of models into account.

If the goal is simply to find the model that predicts best, then this complication does not matter. We place no constraints on either HBRL or LINMAP and choose the model that predicts best. However, we might have other goals. We might want to gather evidence to attempt to infer the decision rule that the respondent is using. In this case, we might take as evidence a finding that the best lexicographic model predicts better than the best compensatory model. In order to gather such evidence, we must define what we mean by compensatory.

In general, compensatory means that doing well on some aspects can compensate for doing poorly on other aspects. The least restrictive version of a non-compensatory decision rule is an additive partworth model in which the largest (aspect-based) partworth exceeds the sum of all other (aspect-based) partworths, plus, the second-largest partworth exceeds the sum of all remaining partworths, and so on to the smallest partworth (Kohli and Jedidi 2006). Unfortunately, imposing constraints based on this least restrictive definition is difficult computationally because such a set of constraints does not imply a convex set.

A simpler and more-realistic set of constraints is to define a q-compensatory model such that no partworth ratio is larger than q. (The largest partworth can be no larger than q times the smallest partworth.) This is an extension of the definition used by Bröder (2000) in psychology. (Bröder's comparison used $q = 1$.) Note that this is a definition. By the principle of optimality, imposing the q-constraints will lead to worse fit, although it might lead to better predictions. Thus, we cannot, in principle, write an algorithm to find the "best" q, unless we use holdout data to identify q, which would not be appropriate.

We compare models on three metrics. Our fit measure is the percent of ranked-pairs that are not violated. This measure is optimized by the greedoid-based dynamic program, although the program is searching over a finite set of orders compared to an uncountable infinite set of partworths that is evaluated by HBRL or LINMAP. Although the details vary, this fit measure is close to that which is optimized by LINMAP. This fit measure is not optimized by HBRL, although we expect that HBRL will do well on this fit measure.

In addition to our fit measure, we compare the performance of the greedoid-based dynamic program and the benchmarks on their ability to predict holdout data. The two measures of prediction are (1) holdout pairs that are not violated and (2) holdout hit rate.

EMPIRICAL DATA

Yee, et. al. tested greedoid methods with a 2x2 empirical experiment in which respondents evaluated 32 smartphones profiles that varied on 16 aspects which were grouped into seven features: carrier (Verizon, Cingular, Sprint, Nextel), manufacturer (Sony, Samsung, Nokia, Blackberry), price ($99, $199, $299, $499), operating system (Palm, Microsoft), form (flip, brick), keyboard (mini, none), and size (small, large). Half the respondents were asked to first indicate which smartphones they would consider and then to rank only those smartphones that they would consider. The other respondents ranked all 32 smartphones. In a full-crossed design, half of the respondents were allowed to presort smartphones by features and half were not. There was also a fifth cell in which respondents were presented with only 16 smartphones in the consider-then-rank, no-sort task.

Screen shots are given in Yee, et. al. Respondents were shown a fractional factorial of 32 profiles. Each profile (smartphones) was represented by icons which illustrated the features. The task and the questionnaire was pretested carefully and respondents understood the task, the icons, and the questions. To indicate consideration, respondents clicked on the smartphone icon. After the click, the considered smartphone was surrounded by a blue box. When respondents were finished indicating their consideration set, they clicked to continue. Smartphones that were not considered disappeared from the screen and only the considered smartphones were displayed. Respondents then clicked on their first choice, which disappeared from the screen. This continued until all considered smartphones had been ranked. Respondents in the rank-only task where not shown the consideration screen; they completed the rank task for all 32 smartphone profiles. In the two experimental cells in which respondents were allowed to presort the smartphone profiles, they were presented with three drop-down boxes in which they could choose features on which to sort. They could sort as many or as few times as they wanted.

In the holdout task, respondents were presented with two sets of four additional smartphone profiles, chosen from a second fractional factorial of the seven features. We designed the holdout task so that it had a different look and feel. In particular, respondents sorted the four profiles with a task similar to that which is used to sort slides in Microsoft PowerPoint. They then indicated which, if any, of the profiles they would consider.[4] Prior to completing the holdout task, and after the initial consider-the-rank or rank-only task, respondents completed a mini-IQ test to cleanse short-term memory (Frederick 2005).

RESULTS (SUMMARIZED FROM YEE, ET. AL. 2006)

Table 1 summarizes the empirical comparisons reported in Yee, et. al. As expected, both LINMAP and the greedoid-based lexicographic-by-aspects (LBA) model do well on fitted pairs, although LINMAP is slightly (but significantly) better. HBRL, which does not optimize fitted pairs, does less well. Constraining the model to be q-compensatory reduces fit, more so for HBRL than for LINMAP. It appears that LINMAP can readjust its optimization to overcome the q-constraint.

[4] For an analysis of the holdout consideration data, see Yee (2006).

Table 1
Fit and Predictive Ability (from Yee, et. al. 2006)

	Fit (Pairs)	Holdout Pairs	Holdout Hit Rate
Lexico by aspects	95.5%	**74.5%***	**59.7%***
Lexico by features	82.8%	65.8%	48.1%
HBRL	87.1%	**74.3%***	54.9%
HBRL (*q*)	82.8%	69.0%	48.9%
LINMAP	**96.9%***	**73.6%***	54.9%
LINMAP (*q*)	95.7%	**73.6%***	**56.9%***

*Best or not significantly different than best at 0.05 level

The more interesting comparison is on the holdout data. LBA does better, albeit not significantly so on holdout pairs, than either HBRL or LINMAP. We are quite impressed with this result, which we interpret as "at least as well," because LBA is a much more highly constrained model than either HBRL or LINMAP. It certainly suggests further investigation.

When we constrain the additive models so that they are truly q-compensatory, then holdout predictions are significantly worse for HBRL (q).[5] One interpretation is that respondents may, in fact, be using simpler non-compensatory decisions. However, this interpretation can at best be treated as initial evidence and is subject to further empirical tests.

The comparison to a LINMAP-based q-compensatory model is quite interesting. Because LINMAP optimizes fitted pairs, there is a danger of over-fitting the data. Various constraints might help mitigate over-fitting. The non-compensatory constraints are one such set of constraints and they seem to improve holdout predictions at the expense of fit.[6] On the other hand, the q-compensatory constraints might also mitigate over-fitting. The improvement is slight relative to an unconstrained LINMAP, and not as much as LBA, but it is interesting and worth future research.

Finally, recall that LBF is nested within the more-general LBA. LBF is another form of constraints and may prevent over-fitting if respondents are truly using features rather than the more-detailed aspects to rank profiles. However, on average, predictions are not improved with LBF (relative to LBA) suggesting that most respondents are likely using aspects rather than features to order profiles. This result is intuitive in this category. Respondents might have a favorite carrier, say Verizon, which becomes an acceptance aspect. Once the consideration set is limited to Verizon, the respondent may prefer to rank on other aspects, say price or form, than rank on the remaining carriers.

---

[5] In this table we use $q = 4$. Yee, et. al. provide results for q that varies from $q = 2$ to $q = \infty$. The qualitative interpretations are relatively insensitive to the choice of q within reasonable ranges.

[6] The greedoid-based dynamic program searches over a finite set of aspect orderings rather than an uncountable infinite set of partworths. For every ordering there is a set of partworths that acts isomorphically.

Yee, et. al. also use the data to address a series of behavioral questions. We summarize the results here:

- the consider-then-rank task is rated significantly better on enjoyment, interest, and perceived accuracy and, for the no-sort cells, takes significantly less time to complete,
- about 2/3rds of the respondents' holdout predictions are at least as good with LBA as with a compensatory model and predictions are, on average, about 5 points higher,
- about 2/3rds of the respondents appear to be using aspects rather than features to sort profiles,
- Yee, et. al. obtain the same pattern of significance with either holdout hit rates or holdout pairs,
- giving respondents the opportunity to presort profiles does not appear to provide a significant difference in respondents' tendency to use lexicography,
- more respondents appear to be lexicographic if they are asked to complete a rank task than a consider-then-rank task,[7]
- for the consider-then-rank task, there does not appear to be a significant difference in the use of lexicography among those respondents who asked to evaluate 32 profiles as compared to 16 profiles.[8]
- when the analysis is replicated on a data set from Lenk, et. al. (1996), Yee, et. al. obtain a similar pattern of results as with the smartphone data. For example, LBA does just as well as HBRL on holdout hit rate. It does well, but not quite as well as HBRL on holdout pairs.[9]

Table 2
Additional Results on LINMAP and the Use of Self-Explicated Data

	Fit (Pairs)	Holdout Pairs	Holdout Hit Rate
Strict LINMAP w/o SEs	**96.9%***	**73.6%***	**54.9%***
Classic LINMAP w/o SEs	79.3%	67.2%	45.5%
Strict LINMAP w/ SEs	88.3%	**72.3%***	**53.1%***
HBRL w/o SEs	87.1%	**74.3%***	**54.9%***
HBRL w/ SEs	79.4%	69.3%	49.9%

*Best or not significantly different than best at 0.05 level.

---

[7] The number of pairs on which the model fit is optimized is less with the consider-then-rank task than with the full-rank task.. Although this applies equally to HBRL, LINMAP, and the greedoid-based dynamic program, we cannot rule out an interaction in the sense that one of the methods might be more sensitive than the others to the amount of information available.

[8] At first glance this null result appears to contradict the standard results in behavioral science such as in Payne, Bettman, and Johnson (1993). However, the majority of the standard results deal with a choice task rather than a consider-then-rank task. The Yee, et. al. data compare the percent of non-compensatory decision making for the consider-then-rank task. Most of the pairs in the consider-then-rank task result from consideration decisions by the respondent. The second rank-phase plays only a small role in estimation.

[9] LINMAP does not do as well as HBRL on the Lenk, et. al. data. The original Lenk, et. al. data were ratings data. HBRL, LINMAP, and the greedoid-based dynamic program are based on degrading the ratings data to retain only rank-order information. Interestingly, the holdout hit rate obtained with LBA and HBRL on the degraded data is not significantly different than the holdout hit rate that Lenk, et. al. obtained using HB on the metric data.

ADDITIONAL RESULTS

Yee, et. al. do not report a comparison of classic LINMAP to the new strict-pairs LINMAP. Nor do they report the details of a comparison in which the estimation methods use self-explicated data as well as the rank (or consider-then-rank) data. These results are presented in Table 2.

As predicted by Srinivasan (1998), the new strict-pairs LINMAP does significantly better than classic LINMAP on fit, holdout pairs, and holdout hit rate.

When self-explicated data (SEs) are added to LINMAP as constraints, the fit measure is degraded as is expected by the principle of optimality. However, LINMAP with SEs does as well as LINMAP without SEs on both holdout pairs and holdout hit rate. As recommended by Sawtooth (Sawtooth Software 2001), we expect HBRL with SEs as constraints to improve predictive ability. This does not seem to be the case with our data. See also discussion in Hauser and Toubia (2005) and Liu, Otter, and Allenby (2005).

SUMMARY

Greedoid-based methods provide a new, practical means to infer non-compensatory decision processes from the data that we normally collect for conjoint analysis. The data collected by Yee, et. al. were based on a full-profile rank task or a full-profile consider-then-rank task. The data collected by Lenk, et. al. (1996) were based on a full-profile ratings task. The greedoid-based dynamic program is also applicable to choice tasks and/or partial-rank tasks. With suitable modification, it should be applicable to partial-profile task.

The greedoid methods and related developments in the inference of non-compensatory decision making are new, but promising. Initial empirical tests suggest that the new methods fit or predict holdout data as well as less-constrained methods (see also Kohli and Jedidi 2006). The methods are easy to use, fast, and feasible for practical numbers of aspects. They can identify the features that respondents “must-have,” that is, features that are ranked highly in a lexicographic aspect order. We are optimistic and hope that other researchers continue to explore non-compensatory inference from conjoint-like data.

Greedoid-based methods are certainly not the only means to analyze non-compensatory decisions. Kohli and Jedidi (2006) used a modified greedy heuristic and report excellent results. Gilbride and Allenby (2004) use an HB specification to infer a screening process and also report excellent results.

REFERENCES

Bröder, Arndt (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, 5, 1332-1346.

Frederick, Shane (2005), "Cognitive Reflection and Decision Making," Journal of Economic Perspectives. 19, 4, (Fall), 24-42

Gilbride, Timothy and Greg M. Allenby (2004), "A Choice Model with Conjunctive, Disjunctive, and Compensatory Screening Rules," Marketing Science, 23, 3, (Summer), 391-406.

Hauser, John R. and Olivier Toubia (2005), "The Impact of Utility Balance and Endogeneity in Conjoint Analysis," Marketing Science, 24, 3, (Summer), 498-507.

Kohli, Rajeev and Kamel Jedidi (2006), "Representation and Inference of Lexicographic Preference Models and Their Variants," Working Paper, Columbia University, New York, NY, April.

Lenk, Peter J., Wayne S. DeSarbo, Paul E. Green, and Martin R. Young (1996), "Hierarchical Bayes Conjoint Analysis: Recovery of Partworth Heterogeneity from Reduced Experimental Designs," Marketing Science, 15, 2, 173-191.

Liu, Qing, Thomas Otter and Greg M. Allenby (2006) "Investigating Endogeneity Bias in Conjoint Models." Working Paper, Ohio State University, Columbus, OH, (January).

Martignon, Laura and Ulrich Hoffrage (2002), "Fast, Frugal, and Fit: Simple Heuristics for Paired Comparisons," Theory and Decision, 52, 29-71.

Payne, John W., James R. Bettman, and Eric J. Johnson (1993), The Adaptive Decision Maker, (Cambridge, UK: Cambridge University Press).

Sawtooth Software (2001), "The ACA/Hierarchical Bayes Technical Paper," (Sequim, WA: Sawtooth Software, Inc.)

Srinivasan, V. (1998), "A Strict Paired Comparison Linear Programming Approach to Nonmetric Conjoint Analysis," in Operations Research: Methods, Models, and Applications, Jay E. Aronson and Stanley Zionts, eds., (Westport, CT: Quorum Books), 6-111.

______ and Allan Shocker (1973), "Linear programming Techniques for Multidimensional Analysis of Preferences, Psychometrika, 38, 3, (September), 337-369.

Tversky, Amos (1972), "Elimination by aspects: A theory of choice," Psychological Review, 79, 281-299.

Yee, Michael (2006), "Inferring Non-Compensatory Choice Heuristics," Ph.D. Thesis, Operations Research Center, MIT, Cambridge, MA. (June)

______, Ely Dahan, John Hauser, and James Orlin (2006), "Greedoid-Based Non-compensatory Two-Stage Consideration-then-Choice Inference," (April) forthcoming, Marketing Science.