ValueCharts: Analyzing Linear Models Expressing Preferences and Evaluations

Giuseppe Carenini and John Lloyd
Department of Computer Science
University of British Columbia
2366 Main Mall, Vancouver,
B.C. Canada V6T 1Z4
+1 604 822-5109
carenini,lloyd@cs.ubc.ca

ABSTRACT

In this paper we propose ValueCharts, a set of visualizations and interactive techniques intended to support decision-makers in inspecting linear models of preferences and evaluation. Linear models are popular decision-making tools for individuals, groups and organizations. In Decision Analysis, they help the decision-maker analyze preferential choices under conflicting objectives. In Economics and the Social Sciences, similar models are devised to rank entities according to an evaluative index of interest. The fundamental goal of building models expressing preferences and evaluations is to help the decision-maker organize all the information relevant to a decision into a structure that can be effectively analyzed. However, as models and their domain of application grow in complexity, model analysis can become a very challenging task. We claim that ValueCharts will make the inspection and application of these models more natural and effective. We support our claim by showing how ValueCharts effectively enable a set of basic tasks that we argue are at the core of analyzing and understanding linear models of preferences and evaluation.

Categories and Subject Descriptors

H.5.2 [User Interfaces]: Graphical user interfaces (GUI); I.3.6 [Methodology and Techniques]: Interaction techniques; H.4.2 [Types of Systems]: Decision support;.

General Terms

Design, Economics, Human Factors.

Keywords:

Information visualization, interactive techniques, linear model, preference model, evaluative index, decision analysis.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
AVI '04, May 25-28, 2004, Gallipoli (LE), Italy
© 2004 ACM 1-58113-867-9/04/0500..$5.00

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
AVI 2004, Gallipoli (LE), Italy
© 2004 ACM 1-58113-867-9/04/0500..$5.00

1. INTRODUCTION

When individuals or groups are faced with complex decisions, they look for a solution that dominates all the others on all the factors they care about. However, in many situations such a solution does not exist, and decision-makers are forced to consider tradeoffs among their objectives (i.e., determine how much they are willing to give up in the achievement of one objective to get a certain increment in another). For instance, if you look for a new house in a given price range, you may not find one that is spacious, has an excellent location and offers great amenities. Therefore, to make your decision you will need to think carefully about your tradeoffs in this domain.

Since people are not naturally very effective at considering tradeoffs among objectives, decision analysis in the last forty years has investigated methods for supporting decision-making with conflicting objectives [8]. Typically, preferences are elicited from the decision-maker to build a quantitative model that includes all the objectives the decision-maker cares about, along with a specification of how the achievement of each objective can be measured. Furthermore, tradeoffs among objectives are quantified in the model by weighting each objective depending on how important it is for the decision-maker. The resulting model (which under reasonable assumptions is linear [9]) can be then applied to any proposed solution to compute its value for the decision-maker.

Decision analysis is not the only field in which linear models are applied to evaluate entities. Similar models are commonly used in Economics and the Social Sciences to rank entities according to an evaluative index of interest. For instance, the UN has developed an index to rank countries according to their level of development[1] (i.e., the Human Development Index (HDI)). Or, as another example, Mercer, a leading consulting company, has developed an index to rank cities according to their quality of living[2]. As a matter of fact, several indexes are published every year to rank a large variety of entities. Individuals, public and private organizations use them to guide their decision-making about policies and interventions.

The key difference between evaluative indexes and preference models from decision theory is that while preference models are decision-maker specific (i.e., a different model is needed for each

[1] http://hdr.undp.org/

[2] http://www.imercer.com/globalcontent/employeemobility/Quality.asp

individual), evaluative indexes are typically generic in nature (i.e., the same model is used by everybody). Nevertheless, evaluative indexes are structurally equivalent to preference models. They are linear models decomposing the value of an entity into measurable factors that are weighted by importance.

The goal of building models of preferences and evaluations is not simply to rank alternatives according to their value. Rather, the fundamental goal is to help the decision-maker organize all the information relevant to generate the ranking into a structure that can be effectively analyzed. The decision-maker should at least be able to analyze (i) the model itself (ii) the final (and partial) results of applying the model to the alternatives (iii) how all these results are sensitive to the model parameters (e.g., weights expressing tradeoffs). As the number of alternatives is typically quite large, and the models tend to be quite complex, the resulting analysis task can be very challenging.

Given the importance and popularity of linear models expressing preferences and evaluations, and the challenges associated with getting insights from their analysis, it is surprising that, with the exception of [1, 17], little attention has been paid to devise effective interactive visualization techniques to support their analysis. The goal of this paper is to propose a new technique that we argue will make the inspection and application of linear models expressing preferences and evaluations more natural and effective. Although at this point we have not performed any empirical evaluation of ValueCharts, in the paper we discuss their potential benefits by referring to a set of basic tasks that any technique with a similar purpose should support.

In the remainder of the paper, we first describe sample linear models, sample datasets and how models are applied to the datasets. Next, we present a set of tasks that any effective technique for analyzing linear models should facilitate. After that, we discuss how ValueCharts have been designed to effectively support these tasks. Then, we review previous work on visualization for decision-making and preference models. Finally, we briefly describe our implementation of ValueCharts and discuss plans for extensions and user studies. A demonstration of ValueCharts can be run from http://www.cs.ubc.ca/~lloyd/java/valueCharts.html (If prompted for a user id and password, please use ubc-vsgl for both).

2. LINEAR MODELS EXPRESSING PREFERENCES AND EVALUATIONS

Preference models for Preferential Choice in Decision Theory: Many complex decisions involve the selection of a preferred alternative out of a set of entities (e.g., objects, courses of action) and require consideration of tradeoffs among multiple conflicting objectives. In these situations, called preferential choices, decision analysis indicates that the decision-maker can benefit from expressing her values and preferences with respect to the entities in a model that under reasonable assumptions is linear. Such a model is called an additive multi-attribute value function (AMVF) and was devised in the context of the most popular decision analysis theory for preferential choice (Multi-Attribute Utility Theory (MAUT) [8]).

An AMVF comprises a value tree and a set of component value functions. A value tree is a decomposition of an entity value into a hierarchy of entity aspects (called objectives in decision theory). See left of Figure 1 for a simple value tree in the real estate domain for the decision-maker *dm*. Each leaf-node of a value tree is referred to as a primitive objective (e.g., *Park-Distance* in Figure 1), while each non-leaf-node is referred to as an abstract objective (e.g., *Location* in Figure 1).

Each primitive objective is associated with a component value function, which expresses the preferability of each domain-value for that objective as a number in the [0,1] interval, with the most preferable domain-value mapped to 1, and the least preferable one to 0. For instance, in Figure 1 (right) the *Eastend Neighborhood* is the most preferred by *dm*, and a *distance-from-park* of 1 mile has preferability (1 - (1/3.2 * 1))=0.69.

The arcs in the value tree are weighted to represent how valuable it would be for the decision maker to move from the worst to the best level of an objective (with respect to do the same for its siblings). For instance (see in Figure 1), *dm* would consider moving from a Southside house 5km from a park (worst *location*) to a Eastend house right on a park (best *location*) more than twice as valuable (i.e., 0.7 vs. 0.3) as going from an house with no deck and no porch to one with the biggest possible deck and porch.

The sum of the weights at each level is always equal to 1.

Formally, an AMVF predicts the value $v(e)$ of an entity e as a linear combination of the values of the primitive objectives:

$v(e) = v(x_1,\ldots,x_n) = \Sigma w_i v_i(x_i)$, where

- $(x_1,\ldots,x_n)$ is the vector of primitive objective values for an entity e
- $\forall$ primitive objective i, v_i is the component value function and w_i is its absolute weight, with $0 \leq w_i \leq 1$ and $\Sigma w_i = 1$; w_i is equal to the product of all the weights on the path from the *root* of the value tree to the primitive objective i.

Notice that the value $v(e)$ of an entity e will be in [0,1], with 1 meaning the best possible e and 0 meaning the worst possible e.

For illustration, Figure 2 shows graphically the application of *dm*'s AMVF to two alternatives: House-A and House-B. From right to left, first the component value functions are applied. Next the computed values are multiplied by the corresponding absolute weights. Finally the value of the higher nodes is computed by summing up the values of their siblings.

AMVFs are popular and effective models that can support individuals, groups as well as public and private organizations in making complex decisions. For examples of complex linear models of preferences in business and public policy see [9] Chapter 11. This chapter provides a thorough description of how these models are elicited and used. An additional intuitive and detailed example of preference model for a small group (a family selecting a house to buy) can be found in [7].

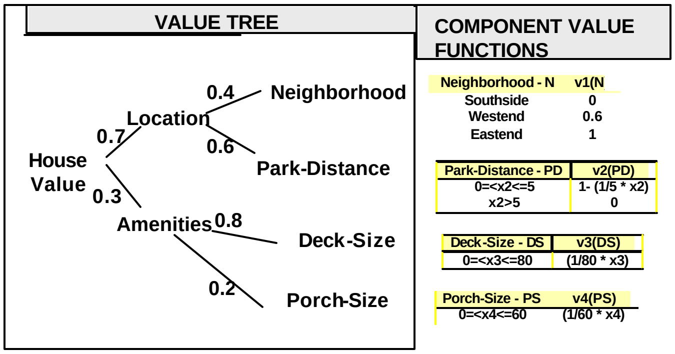


Figure 1 Sample Additive Multi-attribute Value Function (AMVF)

Evaluation Indexes in Politics and Economics:

The use of linear models to evaluate entities is not limited to decision analysis. In Economics and the Social Sciences several

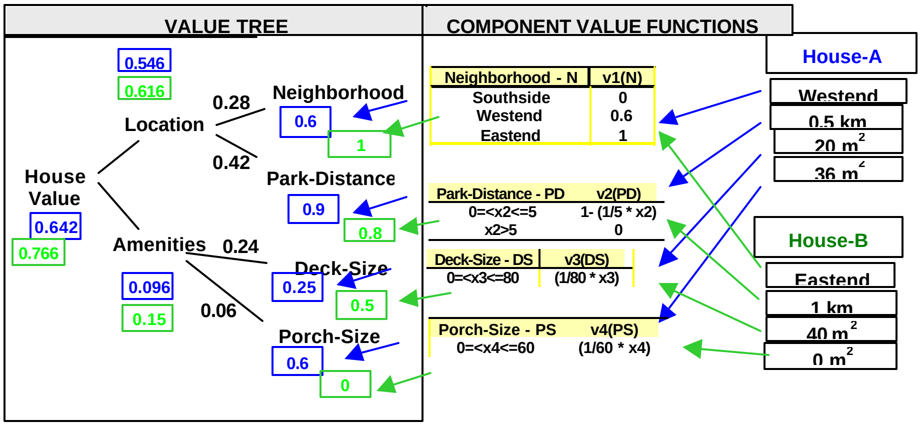


Figure 2 Application of AMVF to two alternatives

indexes are published every year to rank a large variety of entities. These indexes have great influence on decision-making at all levels in our society, as individuals, public and private organizations use them to design and justify their policies and interventions.

Figure 3 shows a graphical depiction of the Human Development Index (HDI), an evaluative index published by the UN to rank countries according to their level of development[3]. As can be seen in the figure, an evaluative index is structurally equivalent to an AMVF. Relevant *indicators* correspond to the AMVF objectives and are also organized in a weighted-tree. Furthermore, the indicator formulas correspond to the AMVF component value functions, by mapping domain-values (e.g., GDP 15,000$) into the [0,1] interval. Similarly to AMVFs, the evaluation of an entity according to an index is computed as a linear combination of the values returned by the *indicator formulas* when they are applied to that entity, and the resulting evaluation will also lie in [0,1].

As we pointed out in the introduction, the key difference between an evaluative index and an AMVF is that while AMVFs are decision-maker specific (i.e., a AMVF is needed for each individual), an evaluative index is typically generic in nature (i.e., the same model is used by everybody).

3.VALUE-CHARTS

The fundamental goal of modeling in supporting preferential choice is not to return an alternative that must be blindly accepted. Rather, it is to provide the decision-maker with insights on the decision problem. In other words, the fundamental goal is to enable the decision-maker to organize all the information relevant to a complex decision into a structure that can be effectively analyzed [5]. Therefore, building linear preference models is only the initial step in selecting the best alternative when multiple conflicting objectives are involved. Once (and while) the model is built, the decision-maker can get insights on the decision problem by effectively analyzing:

– The model itself
– The final (and partial) results of applying the model to the alternatives
– How all these results are sensitive to the model parameters (e.g., weights)[11]

Likewise, when an evaluative index is specified and applied to a set of entities, most decision-makers will not be satisfied if only the evaluation and the ranking of the entities are provided. They would likely get more valuable insights on the evaluations, if they could inspect the index structure, its application to the entities, and perform sensitivity analysis.

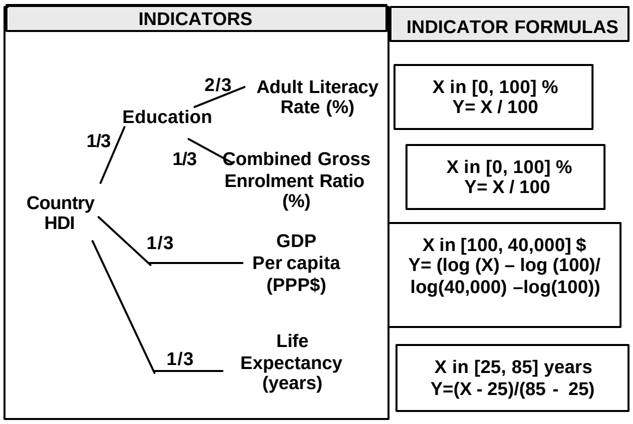


Figure 3 Human Development Index (HDI)

A ValueChart is a set of visualizations and interactive techniques intended to support decision-makers in inspecting linear models of preferences and evaluation. In this section, we first decompose the task of inspecting linear models into a set of simpler conceptual tasks. Then, for each of these tasks, we propose either a visualization or an interactive technique that effectively enables the task. Finally, we show how all these techniques can be seamlessly integrated in a ValueChart. As a general methodology, instead of trying to develop a completely new set of techniques, we tried as much as possible to integrate familiar techniques. Our goal was to design a simple interface for the target complex task

[3] We chose this index for illustration, because it is fairly simple. Evaluative indexes tend to be more complex. For instance, the quality of living index comprises 49 indicators.

of inspecting linear models, aiming at a tool that should be easy to learn and to use for a large user population.

Requirements: Basic Tasks that must be Enabled

Any interface supporting the analysis of linear models expressing preferences and evaluations should enable (and as much as possible facilitate) the following basic conceptual tasks[4] (the tasks are exemplified referring to Figure 2[5]):

1) Comparison of alternatives with respect to total value (e.g., compare the value of HouseA with the value of HouseB)

2) For each alternative, assessment of the contribution to its total value of:
   a) value of each primitive objective (e.g., contribution of Park-Distance to House-Value of HouseA)
   b) value of each abstract objective (e.g., contribution of Location to House-Value of HouseA)

3) Comparison of alternatives with respect to:
   a) value of each primitive objective (e.g., compare the value of HouseA Deck-Size with the value of HouseB Deck-Size)
   b) value of each abstract objective
   c) value across objectives (so values for all objectives need to be expressed in same unit); (e.g., compare the value of HouseA Deck-Size with the value of HouseB Porch-Size)

4) Inspection of the hierarchy of objectives (value tree)

5) Assessment of the extent to which each objective weight contributes to the total (1 when normalized)

6) Sensitivity analysis of changing a weight:
   a) how does it effect other weights? (e.g., if the weight of Deck-Size is changed, how will the weight of Park-Distance change?)
   b) how does it effect the value of the alternatives (both the total and for each objective) ?; (e.g., if the weight of Deck-Size is changed, how will the value of HouseB change? How will the value of Park-Distance for HouseA change?)

7) Inspect component value functions

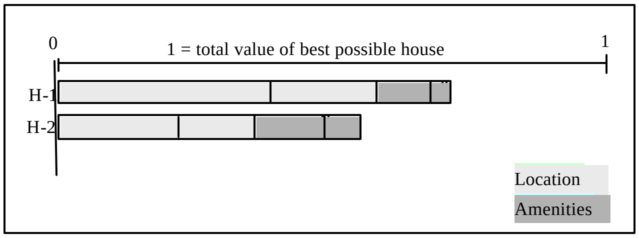
8) Inspect the range on which each primitive objective is defined (e.g., Park-Distance varies between 0 and 5 km)

9) Maintain overview of all the relevant information

Enabling these tasks requires selecting and integrating appropriate visualization and interactive techniques. In our design, we were guided by results from a vast literature on how certain mappings of data properties to graphical properties and certain interactive techniques are more or less effective in supporting given tasks [3, 4, 10]. In our analysis, we will assume that the number of alternatives is typically much greater than the number of objectives/indicators.

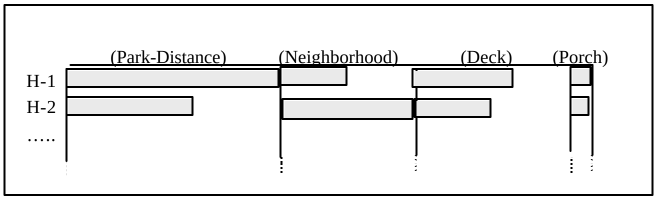
Visualization Techniques

Stacked-bars were selected to enable Tasks 1 and 2. Stacked-bars are the most effective graphical techniques for comparing cumulative totals. Also, 2b can be facilitated by using a combination of proximity, enclosure and color. For instance, the total value of two houses could be displayed as:



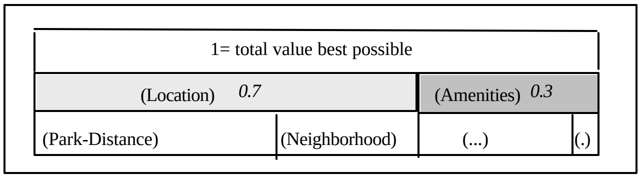
where the "location" objectives are shaded white, while the "amenities" objectives are shaded gray.

For Task 3a, we selected a visualization similar to TableLens [13], in which the rows represent alternatives and the columns represent objectives/indicators. However, in order to enable Task 3c, the amount of space assigned to each column in our table encodes critical information. Each column is assigned an amount of space proportional to the absolute weight of the corresponding objective so that values can be compared across columns. For instance, in the following sample table for the model in Figure 1, the value of H-2 being in the best neighborhood is correctly depicted as 2/3 of the value of H-1 being right on the park.



Task 3b will be enabled by an interactive technique.

Tasks 4 and 5 are enabled by an exploded divided bar chart. Here is an example for the value tree in Figure 1:



The bottom divided chart enables Task 5. The upward explosion enables Task 4 by revealing the hierarchical structure of the value tree.

The remaining tasks can be supported by integrating the visualizations we have presented, so that effective interactive techniques can be applied.

[4] The set we propose is not intended to be exhaustive. Also, the tasks are not ordered by importance. They are numbered only for ease of reference.

[5] Although we use the terminology of preference models in describing the tasks, the same analysis applies to evaluative indexes.

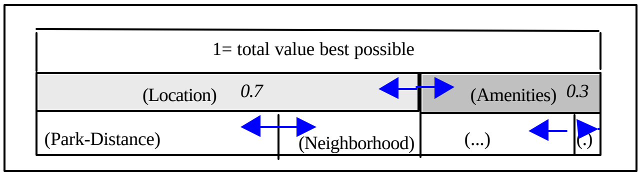
Integration of the Visualizations

The visualizations we have presented so far can be effectively integrated as shown in Figure 4. In this figure, the HDI is applied to 34 countries. Note that:

- The exploded divided chart can be placed on top of the table. The two match nicely because (to enable Task3c) each column in the table is assigned an amount of space proportional to the weight of the corresponding objective. The integration of the hierarchy of objectives with the table effectively enables the interactive technique for sensitivity analysis (as we will see shortly).
- The stacked-bars expressing the cumulative total value are aligned with the table rows and the elements of the stacked bars are ordered like the columns in the table. This facilitates inspection of all the information about each alternative.

Interactive Techniques[6]

Sensitivity analysis of the model weights (Task 6) is enabled by allowing the user to drag the edges of the exploded divided chart to increase/decrease a given weight[7]:



Simultaneously, all the other weights are recomputed accordingly (which enables 6a). Since this operation is also changing the size of the table's columns, the values of all the alternatives for all objectives are appropriately re-scaled. As the stacked-bars for cumulative totals are also recomputed, we have that Task 6b is supported.

Task 3b, the comparison of alternatives with respect to abstract objectives, is enabled by allowing the user to switch objectives at the same level in the exploded divided chart. For instance, if in Figure 4, the user dragged Education to the leftmost position, the table's columns would be reordered accordingly and the same would happen to the elements in the cumulative totals. As a result, the user would be able to effectively compare alternatives with respect to Education (which is an abstract objective). Such a comparison may be further enhanced by applying the sorting techniques we describe next .

Double-clicking the left-mouse button on *any* objective causes the alternatives to be sorted according to their value for that objective. Sorting in this way is a technique we have adopted from TableLens [13] because it allows the user to get a rough idea of the objective value distribution. Also, it can reveal correlation between objectives at any level in the value tree. The ability to switch columns may help here by reducing visual traversal.

By center-mouse clicking on a primitive objective the user can inspect its component value function and its range (Tasks 7 and 8).

The current implementation of ValueCharts offers three additional interactive techniques. First, the name and weight of an objective is displayed every time the cursor moves over the area allotted to the objective in the value tree visualization. This helps in case the space allotted becomes too small to display the objective's name. A second technique allows the user to access the corresponding domain-value by center-mouse clicking on any cell in the objective/alternative table. Finally, the user can filter out alternatives whose value for an objective is below a given threshold value. This threshold value is specified by left-clicking and dragging a "threshold mask" from the left edge of the column corresponding to the objective.

4. RELATED WORK ON SUPPORTING PREFERENTIAL CHOICE AND VISUALIZING PREFERENCES

Several visualization techniques have been proposed and applied to support preferential choice: the selection of a preferred alternative out of a set of entities each described by the values associated with a set of attributes.

In contrast with our approach, most of these proposals allow the user to explore the set of available alternatives (e.g., houses) according to their attributes' domain values (e.g., how far from the park they are) and not according to a model of user's preferences. Typically, the user can only specify her preferences as set of constraints on the attributes' domain values and visually verify what alternatives satisfy those constraints. Examples of these proposals include the HomeFinder, a system based on dynamic queries and intended to support people in finding preferred homes [18], and also, the more recent Attribute Explorer [15], in which a distribution of all objects' domain values over each attribute is displayed and effective techniques for querying and filtering are provided.

In general, although these and similar approaches (e.g., [12, 19]) can effectively support the user in exploring and understanding what alternatives are available and their domain properties, they do not support users in following decision strategies that (according to decision analysis) lead to optimal decisions. In particular, not only are users not forced to explicitly consider and quantify their trade-offs among attributes, but they may even be led into following decision strategies promoting trade-off avoidance [16]. Based on these observations, we argue that these approaches should be used with caution. They may be fine for low-stake decisions (e.g., buying a consumer product). However, for high-stake decisions (e.g., buying a house) they should only be used in combination with a system that also allows the decision-maker to express and analyze a model of her preferences.

VEIL [2] is a system that combines visualizing alternatives according to both their attributes' domain values and to a linear model of the user preferences. The system is intended to support users in selecting round-trip flights. All alternative flights for a given trip are displayed to show their departure/arrival time, duration etc. In addition, each graphical object corresponding to a flight is colored (on a gray-scale) depending on how preferable it

[6] The reader can verify the effects of the interactive techniques on an online prototype of ValueCharts available at:

http://www.cs.ubc.ca/~lloyd/java/valueCharts.html
A browser equipped with Java1.2 is required. And, if access requires login/passwd, please use ubc-vsgl for both.

[7] This is just one way in which weights can be changed. Other interactive techniques to change weights could be devised within the context of an exploded divided chart. For instance the pump technique presented in [1] could be adapted to this purpose.

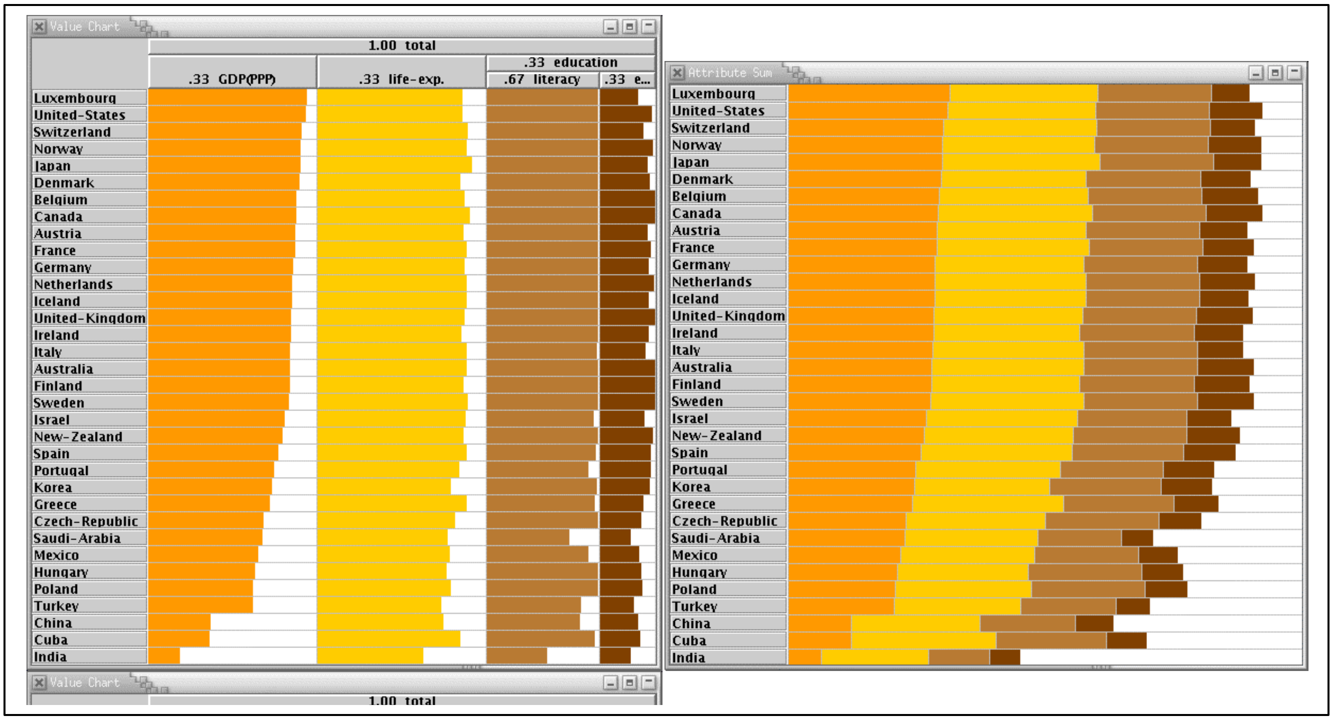


Figure 4 Sample ValueChart: the HDI is applied to a set of countries

is for the user according to a linear model of her preferences. The user can change the preference model in two ways: (i) directly, through a window in which each attribute and its weight is shown and can be changed; (ii) indirectly, by stating a preference on the flights displayed on the screen. With respect to our approach, VEIL allows the user only a minimal inspection of the preference model and of its application to the alternatives. Also, no support for efficient sensitivity analysis is provided.

In the literature, we found only two previous studies of visualization methods for inspecting linear models of preferences and evaluations. [17] presents an interface for preferential choice based on an AMVF in the context of an experiment to test the influence of computer-based decision aids on decision strategy selection. We describe the interface and then briefly discuss how it fares with respect to the set of basic tasks we specified to design ValueCharts. The interface proposed in [17] is sketched in Figure 5. We give here an overview of its essential features and skip unnecessary details. The interface consists of a table in which the rows correspond to objectives and the columns correspond to alternatives (the opposite of ValueCharts). An extra column is added on the right in which the weights are specified. And an extra row is added at the bottom in which the totals are displayed. The table cells contain the score of the corresponding alternative for the corresponding objective on a standardized-scale (i.e., they assume the component value function has already been applied). The user can apply the weights to the whole table to obtain a table of weighted-scores that are then summed up in the total row. In [17] all the cells contain numbers, but for the sake of comparison, we can assume that the cell content, like in ValueCharts, could be display as a bar.

Even with this enhancement, the interface would not fare very well with respect to the tasks we discussed in the previous section. First, it does not express an objective hierarchy and it does not seem that such a hierarchy could be easily added (Tasks 4-5). Second, totals cannot be effectively compared (Task 1). Third, there is no effective display of how each objective contributes to total for each alternative (Tasks 2). In light of this comparison, we argue that ValueCharts are more effective tools for analyzing AMVFs than the interface presented in [17].

[1] proposes a method for the analysis of a preference model that is slightly different form the linear models we have discussed in this paper. The preference model is based on the Analytic Hierarchy Process (AHP) [14], which is a decision-making methodology for preferential choice alternative to MAUT. Although the foundations of AHP have been convincingly criticized in [6], AHP still has some followers. The key difference between an AMVF (or an evaluative index) and an AHP model is the following. While at each leaf of an AMVF we have a component value function mapping the objective domain values in the [0,1] interval, in an AHP model we have instead an additional branching of the value tree. In this branching each alternative is weighted with respect to the others on its value for the corresponding objective (and the weights are normalized to sum 1). As a result, an AHP model nicely maps into a treemap

visualization method. For illustration, Figure 6 shows an AHP version of the HDI applied to three countries and Figure 7 shows how the HDI of these three countries may be visualized as proposed in [1].

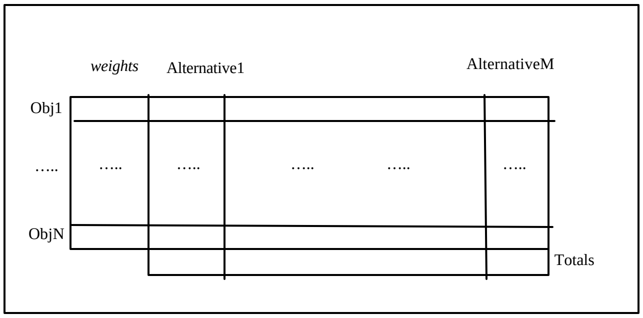


Figure 5 Sketch of the interface for preferential choice presented in [17]

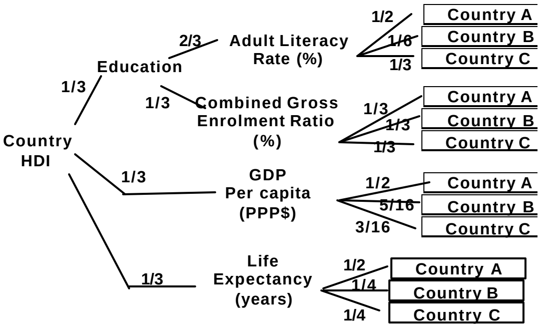


Figure 6: AHP version of the HDI when applied to three countries

In spite of the differences between our linear models and the AHP model some interesting comparisons between ValueCharts and the AHP treemap-based visualization can still be made. As we did before, we can examine how the AHP interface fares with respect to the tasks supported by ValueCharts. The treemap effectively displays the hierarchy of objectives and enables the assessment of their relative importance. It also supports interactive techniques for sensitivity analysis (as described in [1]). However, several key tasks are not efficiently supported. In the treemap, it is extremely difficult to assess the contribution of each objective to the total value of an alternative (Tasks2a-b), because it requires summing and comparing non-adjacent areas. For the same reason, Tasks3a-c (comparing alternatives with respect to an objective and across objectives) is poorly supported. Also notice that these problems become more pronounced as the number of alternatives increases, because the information that needs to be operated upon is further scattered on the treemap.

In conclusion, it seems that ValueCharts could also be effectively used to analyze AHP models.

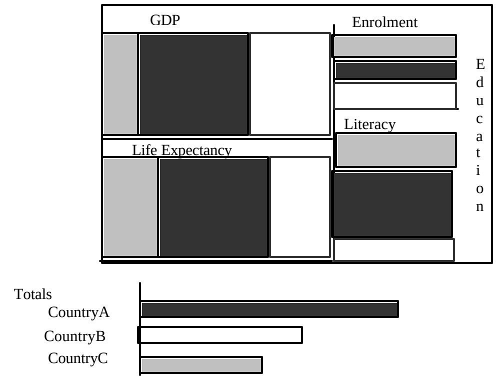


Figure 7 Treemap-based visualization of the model shown in Figure 6

5. PROTOTYPE IMPLEMENTATION

A prototype of ValueCharts has been implemented in Java 1.2. Java was chosen because of its excellent GUI support and ability to create web-based applets. In the present implementation, all information defining the value tree, component value functions, and alternatives is specified in a user-editable specification file. Value tree information is specified first, using a textual hierarchy that reflects the tree structure itself. Component value functions are defined in the value tree leaf-node entries. Two types of functions are presently implemented: a discrete mapping of symbols onto [0,1], and a continuous piecewise-linear mapping of a domain interval onto [0,1].

The contents of the specification file are then read by a Java applet and used to construct the ValueCharts GUI. The GUI itself is built from subclassed Java Swing components, with separate components used for the alternative value display and the various panels representing objectives/indicators. Mouse manipulation is implemented using mouse listeners attached to the relevant component. Only very basic components (such as JLabel, JLayeredPane, JPanel, and JTextField) are used and/or subclassed, and components which display bar-graph information utilize their own paint method. We considered using the more elaborate swing component JTable, but it turned out to be not sufficiently flexible for our purposes.

A demonstration of our prototype can be run from http://www.cs.ubc.ca/~lloyd/java/valueCharts.html (If prompted for a user id and password, please use ubc-vsgl for both).

6. CONCLUSIONS AND FUTURE WORK

Linear models of preferences and evaluation can be effective decision-making tools. They can help the decision-maker organize all the information relevant to a decision into a structure that can be effectively analyzed. However, as these models and their domain of application grow in complexity, their analysis can overburden decision-makers. To address this problem, we have proposed ValueCharts, a set of visualizations and interactive techniques intended to support decision-makers in inspecting linear models. In the paper, we have first identified a set of basic tasks at the core of analyzing liner models, and then we have shown how ValueCharts are superior to previous proposals in effectively supporting those tasks. Although at this point we have not performed any empirical evaluation, we argue that a key strength of ValueCharts derives from their simplicity. As they coherently integrate rather simple visualization and interactive techniques, they should be easy to learn and use.

Several aspects of ValueCharts need further study. In their current implementation, because of the limited display size, ValueCharts support the analysis of relatively simple models (<= 20 objectives) when applied to rather small set of alternatives (<= 50). We plan to investigate how ValueCharts can be scaled-up to larger models[8] and larger datasets. We envision that techniques from TableLens might help with the latter problem, while the former might require completely novel techniques.

Another issue we intend to explore is how ValueCharts could be integrated with techniques that allow the user to explore the set of available alternatives according to their attributes' domain values (discussed in Section 4). In this context, we are particularly interested in the support of "what if?" analysis. For instance, we might allow the user to increase/decrease domain values of an alternative to explore how that changes its evaluation.

ValueCharts have been described in this paper as a tool to inspect linear models of preference and evaluation. We plan to investigate whether ValueCharts could also support the construction of such models. As a more long-term goal, we intend to study how ideas developed for ValueCharts can be extended to devise techniques for non-liner models.

Finally, in order to test ValueCharts and the extensions we have just discussed, we plan to perform user studies. To test ValueCharts on preference models, we envision our subjects performing preferential choices in realistic domains (e.g., real estate). To test ValueCharts on evaluative indexes, we envision our subjects performing analysis tasks on the index application (e.g., find and explain the main changes in the HDI ranking if the weight of GDP is reduced by a given amount).

References

[1] Asahi, T., D. Turo, and B. Shneiderman, *Using Treemaps to visualize the Analytic Hierarchy Process.* Information Systems Research, 1995. 6(4): p. 357-375.

[2] Blythe, J. *Visual Exploration and Incremental Utility Elicitation*. in *AAAI-02*. 2002.

[3] Bertin, J., *Semiology of Graphics.* 1983.

[4] Card, S.K., J.D. Mackinlay, and B. Shneiderman, *Readings in Information Visualization - Using Vision to Think*. 1999: Morgan Kaufmann.

[5] Clemen, R.T., *Making Hard Decisions: an introduction to decision analysis*. 1996, Belmont, California: Duxbury Press.

[6] Dyer, J.S., *Remarks on the Analytic Hierarchy Process.* Management Science, 1990. 36(3): p. 249-258.

[7] Hammond, J.S., R. Keeney, and H. Raiffa, *Smart Choices: a practical guide to making better decisions.* 1999: Harvard Business School Press.

[8] Keeney, R. and H. Raiffa, *Decision with multiple objectives: Preferences and value tradeoffs.* 1976: Cambridge University Press.

[9] Keeney, R.L., *Value-Focused Thinking: A Path to Creative Decision Making*. 1992: Harvard University Press.

[10] Kosslyn, S.M., *Elements of Graph design.* 1994.

[11] Payne, J.W., J.R. Bettman, and D.A. Schkade, *Measuring Constructed Preference: Towards a Building Code.* Journal of Risk and Uncertainty, 1999. 19(1): p. 243-270.

[12] Pu, P. and B. Faltings. *Enriching buyers' experiences: the SmartClient approach*. in *CHI*. 2000. The Hague, Amsterdam: ACM.

[13] Rao, R. and S.K. Card, *The Table Lens: Merging Graphical and Symbolic Representation in an Interactive Focus+Context Visualization for tabular Information*, in *Proceedings of the CHI '94 Conference*. 1994: Boston, MA, USA. p. 318-322.

[14] Saaty, T.L., *The Analytic Hierarchy Process.* 1980, New York: McGraw-Hill.

[15] Spence, R. and L. Tweedie, *The Attribute Explorer: information synthesis via exploration.* Interacting with Computers, 1998. 11: p. 137-146.

[16] Todd, P. and I. Benbasat, *An Experimental investigation of the Impact of Computer Based Decision Aids on Decision making Strategies.* Information Systems Research, 1991. **2**(2): p. 87-115.

[17] Todd, P. and I. Benbasat, *Inducing Compensatory Information Processing through Decision Aids that Facilitate Effort Reduction: An Experimental Assessment.* Journal of Behavioral Decision Making, 2000. 13: p. 91-106.

[18] Williamson, C. and B. Shneiderman. *The Dynamic HomeFinder: Evaluating Dynamic Queries in a Real-Estate Information Exploration System*. in *ACM SIGIR*. 1992.

[19] Wittenburg, K., T. Lanning, M. Heinrichs, and M. Stanton. *Parallel Bargrams for Consumer-based Information Exploration and Choice.* in *UIST*. 2001. Orlando, FL.

[8] For instance the World Competitiveness evaluative index comprises 341 criteria (http://www01.imd.ch/wcy/).