New MaxDiff Designer (V8.1) Offers Small Improvement for HB Estimation

Bryan Orme, Sawtooth Software

Version 8.1 of SSI Web includes an improved experimental designer for MaxDiff studies. This article describes how the new approach obtains superior results in terms of within-version item balance and quantifies the benefits if conducting individual-level analysis (HB). We assume the reader is already familiar with MaxDiff (Maximum Difference Scaling), also known as best-worst scaling, as well as HB estimation.

Introduction

A MaxDiff experimental designer chooses the combinations of items to show within each set, across typically multiple versions (blocks) of the questionnaire. Sawtooth Software's previous version of the MaxDiff designer created nearly perfect designs when considering the aggregate (pooled) information, across all sets and versions. When item prohibitions were not in force, the designs were nearly perfect in terms of:

- One-way item balance (how many times each item occurred)
- Two-way item balance (how many times each item occurred with each other item)
- Positional balance (how many times each item appeared in each position across the sets; 1st position, 2nd position, etc. as seen by the respondents). Positional balance has no effect on statistical precision, but is seen as a way to control for psychological order effects.

Previous versions of our MaxDiff software, however, did not explicitly consider (and therefore did not optimize) the one-way and two-way frequencies *within each questionnaire version*. When we originally developed the MaxDiff designer, researchers tended to focus more on aggregate analysis (logit and latent class) for MaxDiff experiments, and the notion of individual-level analysis (such as via HB) was relatively novel. As a result, our emphasis on achieving optimal designs for aggregate analysis did not optimize within-version item balance. For example, for a symmetric design in which each item could have appeared exactly three times for each version, some respondents may have seen a level two times and others four times. For individual-level analysis (HB), such designs are not as efficient as they could be.

Improved Designer for SSI Web v8.1

With version 8.1, we have modified MaxDiff's design algorithm to emphasize within-version balance, both for one-way and two-way occurrence of levels. We achieved this through adjusting the existing algorithm and by adding a relabeling procedure. The relabeling procedure examines each version of the questionnaire and determines whether for a specific task an item that is shown more times than average should be relabeled as an item that currently is shown fewer times than average. (For example, an item #5 within a specific questionnaire version and specific task could be changed to an item #3.) The relabelings that can best improve the two-way frequency balance are implemented. Upon reviewing the results of the new design procedure, we were especially pleased that the newer designs did not sacrifice any aggregate (across-version) balance in either one-way or two-way frequencies, while achieving substantial within-version improvements on those same measures. This

outcome would suggest that the designs are just as good for aggregate analysis as with the previous algorithm, but are improved with respect to individual-level estimation. We empirically test this in the next sections.

Simulated Datasets Estimated via HB

Improving both the one- and two-way frequency balance within each version is one thing; but achieving more precise HB estimation is the critical matter. Would we see improved results? We used synthetic (computer-generated) data to quantify the gains in individual-level utility precision. Because our aim was to improve statistical measures, there was no need to conduct a study with real respondents. Simulated data actually work better for this type of investigation, as the many unpredictable human factors would not detract from the focus of our research.

Fist, we generated known (true) utilities for 12 items for 300 respondents. The utilities were generated from a population vector of scores arranged in equal increments across the range of -5 to +5, perturbed for each respondent using random (normal) draws with mean 0, and variance 2. These computer-generated respondents were "given" MaxDiff questionnaires resulting from either the previous or the new MaxDiff designer. The robotic respondents "selected" the best item from each set according to their true utilities perturbed by right-skewed Gumbel error. The choice of worst item within each set was selected according to their true utilities perturbed by left-skewed Gumbel error. We generated a total of four different data sets and performed HB estimation on each:

1. **Previous Designer, Asymmetric Design**. 12 items, 7 sets per version, showing 5 items per set. It was impossible to show each item an equal number of times per version. Each item appeared on average 2.92 times.
2. **V8.1, Asymmetric Design** (same design as the previous, but using the new design algorithm).
3. **Previous Designer, Symmetric Design**. 12 items, 9 sets per version, showing 4 items per set. Each item could be shown exactly 3 times per version (though the previous designer often failed to do so).
4. **V8.1, Symmetric Design** (same design as the previous, but using the new design algorithm, which was able to achieve perfect one-way level balance within each version).

Although our synthetic data sets used fewer items (12) than are typically used in practice, we think the results should generalize to situations with more items. The important issue for HB estimation and MaxDiff is how many times each item appears per respondent. Our choice of about 3 occurrences per respondent is fairly typical in the Sawtooth Software community, if robust individual-level scores are required, following recommendations given in our white papers and the MaxDiff documentation.

We analyzed the four synthetic datasets using the HB routine built into SSI Web, with default settings. To ensure that our results were not affected by differences in scale factor (tendency for estimated parameters to uniformly expand or shrink depending on respondent-specific error), we normalized (separately) both the true and the estimated utilities for each individual, so that the mean was 0 and the range was 10. Next, we compared the HB-estimated utilities for each of the 300 respondents to the true. For each respondent and each item, we computed the absolute difference between the true and estimated utilities. We averaged those differences across respondents

and items to create a single estimate of precision, the Mean Absolute Error (MAE)[1]. The results for the four cells appear in Table 1, where smaller MAE means lower error and better precision.

Precision of HB Estimates (MAE)		
	Asymmetric Design	Symmetric Design
Previous Designer	0.8091	0.7804
V8.1	0.7777	0.7584

Table 1

For both asymmetric and symmetric designs, the new designer (V8.1) achieved smaller errors than the previous experimental designer. The efficiency of one design relative to another (concerning individual-level parameter recovery) can be computed by taking the ratio of the squares of these MAE values. As shown in Table 2, for individual-level estimation via HB, the older design routine is about 92 to 94% as efficient as the v8.1 designer.

Relative Individual-Level Design Efficiency Previous Designer Relative to V8.1	
Asymmetric Design	Symmetric Design
$(0.7777^2) / (0.8091^2) = 0.924$	$(0.7584^2) / (0.7804^2) = 0.944$

Table 2

For MaxDiff studies where individual-level results are key, then the new designer would appear to offer modest practical benefits. Examples include:

- Segmentation studies based on latent class or cluster analysis on normalized HB scores, where the researcher is concerned about assigning respondents reliably into different segments.
- Individual predictions of item choice, such as resulting from what-if simulators.
- TURF (Total Unduplicated Reach and Frequency) optimization simulations, where the goal is to find optimal portfolios of items that "reach" respondents.

Having better item balance and more efficient designs at the individual level makes the individual-level estimates more uniformly precise across the items. From an aggregate estimation standpoint, improving the individual-level balance might offer an even tinier improvement as well for population estimates (such as via aggregate logit), though we did not investigate that here.

Conclusion

We think these findings should be viewed as good news, and more good news: the new MaxDiff experimental designer offers a small performance benefit for HB estimation (now the most common approach among our users), and the older version of the designer was already doing quite well (despite the lack of emphasis on within-version level balance). Those who worry enough to count how many times each item appears for each respondent will be satisfied that there is better balance than before, and the statistical results are slightly improved.

[1] We also computed the error in terms of RMSE (Root Mean Square Error), and came to essentially the same conclusions regarding relative error of designs. Because of the ease of describing MAE, we report MAE here.