Sawtooth Software

RESEARCH PAPER SERIES

Three Ways to Treat Overall Price in Conjoint Analysis

Bryan Orme
Sawtooth Software, Inc.

Three Ways to Treat Overall Price in Conjoint Analysis[1]

[2]Bryan Orme, Sawtooth Software
April 2025

This article discusses three ways to treat overall price in conjoint analysis studies:

- Traditional Approach
- Conditional Price (and the related alternative-specific price approach)
- Summed Price

The traditional approach is the easiest to manage, but the other two techniques offer benefits for more advanced applications in specialized situations. I don't know the history of the conditional price and summed price approaches, but they probably were being used as early as the 1970s. My first experiences with them occurred in 1993. Because our work with Adaptive CBC used the Summed Price approach ("A New Approach to Adaptive CBC," Johnson and Orme 2007), it was important to investigate the stability of price estimates under summed price. Simulation results are reported at the end of this article.

Traditional Approach, with Price as Separate Attribute

The typical approach to price is to include it as a separate and independent attribute in the study design. For example, if we were studying laptop computers, we might include the following attributes:

Dell
Acer
Lenovo

4 GB RAM
16 GB RAM
32 GB RAM

500 GB Hard Drive
1 TB Hard Drive
2 TB Hard Drive

[1] This 2025 version of this document is an update of the original 2007 version. Main changes involve the use of VIF (Variance Inflation Factor) to stress test choice of random shock in summed pricing experiments, a mention of alternative-specific designs, and replacing the term "continuous price" with "summed price".

[2] Many thanks to Rich Johnson who reviewed earlier drafts of the 2007 version of this document and provided many useful suggestions for improvement.

2.0 GHz Processor
2.5 GHz Processor
3.0 GHz Processor

$700
$1,000
$1,500

With this traditional approach, we vary each attribute independently of the others (an orthogonal design). Level balance is achieved if each level within each attribute appears an equal number of times. Designs that are both level balanced and orthogonal are optimally efficient for estimating the part-worth utilities with precision (assuming respondents answer using a simple additive model). In the example above, we could estimate part-worth utility values corresponding to each of the three levels of price (a part-worth utility function), or we could estimate a single linear term to reflect the slope of price (a vector utility function). Most researchers choose the part-worth utility function, because it is more flexible and can account for non-linearities in the price function. However, it comes at the cost of increased parameters to estimate.

Despite the robust statistical qualities of orthogonal designs, some researchers and respondents have been bothered that product concepts with the best features sometimes are shown at the lowest prices (and products with the worst features are sometimes shown at the highest prices). These combinations seem illogical and often lead to obvious (dominated) choices in the questionnaire. Such questions, while still informative from a statistical standpoint, lead to a less realistic experience for the respondent.

Conditional Pricing

Conditional pricing increases the realism of conjoint tasks. With conditional pricing, incremental amounts are added for premium brands or features, so enhanced products are generally shown at higher prices. We still treat price as a separate attribute with just a few levels (such as three to five). But, those levels of price are described with different absolute dollar amounts, depending on product characteristics. Probably the most common use among Sawtooth Software users is to associate different brands or different brand/package size combinations with different price ranges. For example, the premium brand might be shown at $10, $15, or $20 whereas the discount brand is shown at half those prices: $5, $7.50, or $10. In the design matrix, we still treat price as a single attribute with three levels, even though we display a larger number of actual prices.

Conditional pricing uses a price lookup table to determine actual prices to show in the questionnaire, based on the characteristics of each product. To create the lookup table, we first decide how many attributes will participate in the conditional pricing relationship. We recommend conditional price to be triggered by the attribute or attributes that most drive(s) the total price of the product, and as few attributes as possible, to keep things simpler (with more complex relationships relying on the summed pricing approach covered later in this white paper).

Let’s assume with our previous laptop PC example that we wanted to make price conditional on RAM and Hard Drive. We first start by choosing price premiums associated with those attribute levels. We don’t show these premiums to respondents next to each attribute level, but just use them to determine the overall average price. Only a single total price is shown within the product concept.

Example 2: Conditional Price

Base Price: $750

Dell
Acer
Lenovo

4 GB RAM	+$0
16 GB RAM	+$100
32 GB RAM	+$200

500 GB Hard Drive	+$0
1 TB Hard Drive	+$200
2 TB Hard Drive	+$400

2.0 GHz Processor
2.5 GHz Processor
3.0 GHz Processor

Low Price (-30%)
Medium Price (Average Price)
High Price (+30%)

Let’s assume that the base price for the laptop is $750. We construct a look-up table to determine the prices that we should show on the screen for each possible product combination at each of the three prices. This table would have a total of 3 x 3 x 3 = 27 rows. The first five rows of the price lookup table look like:

RAM	Hard Drive	Price	Text to Display
1	1	1	$525
1	1	2	$750
1	1	3	$975
1	2	1	$665
1	2	2	$950
…	…	…	…

For example, row four of the table specifies what price should be displayed when a product with 4 GB RAM and 1 TB Hard Drive appears at the low price. The price to show is $665. This is determined by taking the base price ($750) plus the price increments associated with the two conditional attributes, and then reduced by 30%.

The benefit of conditional pricing is we display more reasonable prices to respondents and in the simplest case price may be estimated using main effects for (in this example) the three levels of price in the design. And, critically, the design is still orthogonal and unencumbered by prohibitions.

There are a few challenges when working with conditional pricing:

- We no longer can interpret the main effect utilities for attributes involved in the conditional relationship independent of price. For example, we cannot interpret the levels of RAM as the preference for each of its levels holding everything else constant. The utility of each level of RAM is confounded with the incremental price attached to that level. The levels must therefore be interpreted as the preference for levels of RAM given the average prices shown for those levels. So, it's very possible to achieve a higher average utility for 4 GB @ +$0 than 32 GB @ +$400, if respondents on average did not feel that it was worth the extra $400 to have the greater RAM.
- The estimation of part-worth utilities works well when the prices shown to respondents are based on a certain percentage increase or decrease from the average price. However, the resulting prices often need to be rounded to the nearest $100 (or made to end in a "9" for consumer packaged goods). Quite small relative changes in price to round to a more presentable number don't pose much problem. But, significant price changes due to rounding introduce error in the utility estimation for the price attribute.
- If the conditional pricing table is not built in a consistent, proportional manner as specified here (or if rounding resulted in significant deviations from the original formula-based values), it may become impossible to model the data correctly using the conditional price approach unless imposing interaction effects. Interaction effects may lead to overfitting.

Note: alternative-specific (nested attribute) designs are another way to implement conditional pricing. For example, if different brands are typically sold at different prices, then we create multiple price attributes. For example, Brand A might be shown at $10, $12, $15; Brand B at $15, $20, and $25; and Brand C at $25, $35, $45. We create three alternative-specific price attributes each with three levels. The model and simulation predictions for this approach are mathematically identical to conditional pricing with a lookup table triggered only on brand, assuming we add the interaction effect between brand and price to the model. Alternative-specific pricing allows us more flexibility with the prices we set for each brand not needing to be consistent and proportional. The sacrifice for that flexibility is the additional parameters to be estimated in the model with alternative-specific effects.

Summed Price

Another approach offered first in Sawtooth's ACBC software is summed price. (Until it is formally supported in CBC, which may come soon, there are power tricks for implementing it in CBC as well.) Summed price differs from conditional price in two ways. First, it generalizes the idea of conditional pricing (extending it to n-way attributes). Second, it estimates the effect of overall price as a continuous function: typically linear, log-linear, or piecewise. As with conditional pricing, we approach the problem by considering a base price for the product as well as fixed price premiums for levels of non-price attributes (plus or minus some overall independent price variation). If we consider the example from the previous section, the base price is $750 and the most expensive product option would be $1,350 (prior to varying price by some independent amount).

As with the first two pricing approaches, we also only show a single overall price within the product concept, rather than showing prices attached to each attribute level. The only difference between conditional price and summed price is in the coding of the design matrix, where price is coded in a single column as a continuous variable (or as multiple columns in the case of a piecewise function). With the simplest modeling approach, a single price coefficient is estimated based on linear price (or the natural log of price)[3].

Because values in the price column of the design matrix are determined from information in other columns of the design matrix, that column would be linearly dependent on other columns if we didn't do something to break up that dependence. We do this by adding random variation (random shock) to the prices.

The benefits of summed price relative to conditional pricing include:

- In contrast to conditional pricing, the utility of each feature level is estimated independently of any price premium associated with the level[4]. Thus, we would expect the utilities for levels of processor speed to look just like they would when using the standard conjoint approach with no conditional pricing[5]. See Appendix A for more in-depth explanation.
- Since we are estimating price as a continuous function, there is no worry about whether rounding prices to the nearest "9" or the nearest $100 will lead to errors in fitting the data.

But, these benefits come with a serious potential drawback to be managed: the price attribute is positively correlated with any attributes that involve incremental prices in the study, leading to less precise estimates of all effects, but most especially the price coefficient. The amount of correlation among attributes depends on the magnitude of the random shock in overall price as well as the size of the base fixed component of price relative to the incremental prices associated

---

[4] Because the non-price attribute levels are estimated independently of both overall price and any incremental pricing attached to the product features, it shouldn't matter what incremental price levels are attached to each level (within reason, of course). Preliminary evidence from a recent split sample CBC study we conducted supports this.

[5] For evidence of this, see the part-worth utility results for a split-sample study comparing traditional pricing and summed price in Johnson and Orme, 2007.

with each feature level. In the worst case, with no random variation, summed price is simply the sum of the prices associated with the attribute levels. In that case, price would be perfectly predicted by a linear combination of the attributes and the design would be deficient. But, if we additionally vary the overall price by a large enough random amount (see guidelines further below), we can obtain sufficient precision of the estimates for overall price sensitivity as well as the other features in the study.

The final section of this paper includes a simulation study to investigate how much independent variation (random shock) should be specified in the overall price attribute to lead to reasonable estimates with summed price.

Simulation Study

As mentioned earlier, the amount of random shock applied to summed price has a direct impact on the efficiency of the estimates. To provide guidelines regarding how much random variation in price we should include in continuous price designs, for the original 2007 version of this white paper we conducted a synthetic study with 300 simulated respondents and examined the standard error of a linear price coefficient estimate. There were five attributes, each with three levels, along with summed overall price.

For the 2025 update of this white paper, we leveraged a simpler simulation approach (simulating 3000 randomly drawn product alternatives with their summed prices) analyzed with the commonly used VIF (Variance Inflation Factor) calculation. This simpler approach found nearly identical results as the original 2007 simulation study, with less complication. VIF follows from multiple OLS regression, where the dependent variable is summed price and the independent variables are the other non-price attributes in the study, dummy- or effects-coded. VIF for an independent variable in the experimental design is simply 1/(1-Rsquared) from a multiple OLS regression, where the independent variable of interest is regressed on the remaining independent variables.

If the base price is relatively large compared to the incremental prices attached to upgraded features, then (after perturbing overall price by the random price shock) the resulting price attribute will be relatively uncorrelated with the linear combination of the other attributes (lower VIF). However, if the base price is relatively small (or zero!) compared to the incremental prices for the other levels in the study, then the resulting overall price will be more strongly correlated with a linear combination of the other attributes (higher VIF). Therefore, we consider how different relative sizes of the fixed base price relative to incremental prices would affect the results[6].

[6] Another technical issue is that the relative magnitude of the incremental prices for attributes affects the VIF of the summed price attribute. If one attribute contributes disproportionately more to the summed price, the VIF increases. In the 2025 simulations update, we assumed one of the attributes contributes triple the price increment relative to the other attributes.

Simulation Procedure and Results
We simulated different amounts of independent price variation on summed price, from as low as +/-10% to as much as +/-40%. In the original 2007 version of this simulation, we estimated price as a single coefficient, to be applied to the natural log of total price. In the 2025 update to this white paper, we found that the +/-% random shock that we recommended in 2007 (to achieve precision about 50% as efficient for the price coefficient as the 3-level attributes in the same study) was associated with a VIF of about 3.5. Given that guideline, cells in green below are in a "safe" zone for researchers. Cells in red are in the "danger" zone.

VIF (Variance Inflation Factor)

		Random Shock to Summed Price				
		+/- 10%	+/- 15%	+/- 20%	+/- 30%	+/- 40%
Base Price	0 of Average Price	**33.51**	**16.18**	**9.89**	**4.81**	**3.27**
	1/3 of Average Price	**16.77**	**8.23**	**5.01**	**2.84**	**2.07**
	1/2 of Average Price	**10.11**	**5.07**	**3.31**	**2.04**	**1.65**
	3/4 of Average Price	**3.23**	**2.08**	**1.58**	**1.23**	**1.15**

Recommendations
The precision of the utility estimate for summed price depends strongly on both the amount of added random shock and the size of the constant base price relative to the total average price of the product concepts. When the base price of the product is 3/4 the total average price[7], as little as +/-10% price variation on summed price achieves precision of estimates for the price function nearly 50% as efficient as a standard 3-level attribute coded as a part-worth function. However, if the base price is zero (all of the price is explained by the incremental feature prices), then we'd need to vary summed price +/-40% to achieve similar precision. Based on this simulation study, we make the following general recommendations for summed price:

Recommended Minimum Random Shock to Summed Price

If base price is 3/4 of total average price:	+/-10%
If base price is 1/2 of total average price:	+/-20%
If base price is 1/3 of total average price:	+/-30%
If base price is 0 of total average price:	+/-40%

Of course, choosing the amount of random price shock also depends on the client's needs and the market simulations to be run. You should avoid extrapolating beyond the total range of price

[7] It may be useful at this point to illustrate how to distinguish fixed base price from incremental prices for the purposes of applying our results to future studies. Consider an attribute such as brand with 3 levels and incremental prices of Dell +$700, HP +$800, and Toshiba +$900. The fixed component (irrespective of brand) is $700, and the incremental prices are Dell +$0, HP +$100, and Toshiba +$200. Thus, when decomposing price into fixed versus incremental prices, you should ensure that there is no fixed component "hiding" within what appear to be incremental prices. The lowest incremental price for each attribute should start with +$0.

included in the questionnaire. Increasing the random price variation will improve your ability to simulate extreme priced products, at the risk of making the questionnaire present products that seem to have unreasonable prices, given their features.

Appendix A

In this section, I'll describe how coding the overall price attribute as a single column in the design matrix (estimating a single coefficient for price) makes it possible to interpret the utilities of features that have incremental prices associated with them independently of those incremental prices. Consider the simplest of conditional pricing studies, where price is conditional on just one other attribute (brand). Imagine a study with three brands, with prices to be shown as follows:

	Low Price (-30%)	**Middle Price (+0%)**	**High Price (+30%)**
Brand1	$7.00	$10.00	$13.00
Brand2	$10.50	$15.00	$19.50
Brand3	$14.00	$20.00	$26.00

With the conditional pricing approach, we treat the price attribute as a categorical attribute with three levels, -30%, +0%, and +30%. So, if we presented two product alternatives (each at their lowest prices):

Brand1 @ $7.00
Brand2 @$10.50

We would code the alternatives as follows (level 3 is the reference level, and is therefore omitted from the design to avoid linear dependency):

Brand1	Brand2	Price1	Price2
1	0	1	0
0	1	1	0

With this coding scheme, the effect of price is captured as dummy (or effects) coded parameters. Level one is the utility associated with a 30% reduction in price for the given brand (irrespective of the absolute value of price). So, even though the two prices shown to respondents are different ($7.00 for Brand1 vs. $10.50 for Brand2), the coding of price in the design matrix is identical. To account for the fact that there is a price difference in $3.50 between those two product concepts, the disutility for the extra $3.50 in price must be accounted for in the effect for Brand1 relative to Brand2. Thus, the price dummies account for the slope of the curve, and the brand dummies account for inherent preferences for the brands plus the average amount of price difference between prices shown for the brands. Thus, we cannot interpret the brand effects separate from the price premiums reflected by those brands.

However, consider what happens when we code price as a single linear coefficient, as with the Summed Price approach. The two product alternatives are coded as:

Brand1	Brand2	Price
1	0	$7.50
0	1	$10.50

Thus, the information contained in the Price column accounts not only for the relative differences in price, but also that one brand is consistently shown at different prices than the other. The Price coefficient partials out all the effect of price (both the slope and the shift component), and the effects for Brand are captured independently of price. The utilities for brand may be interpreted as their inherent desirability, in the traditional manner of interpreting main effects.