

Tutorial #3: Brand Pricing Experiment

A popular application of discrete choice modeling is to simulate how market share changes when the price of a brand changes and when the price of a competitive brand changes. With latent class choice modeling, it is possible to estimate these changes separately among those who are price sensitive, and those who are not so price sensitive. That is, separate effects can be obtained for each segment as well as the overall market.

One of the flexible features of the 3-file format in Latent GOLD Choice is that it is easy to define the effects and interactions to be included in a model. In this tutorial, we first show how to estimate the effects of brand and price using a model where PRICE and BRAND are treated as a distinct attributes – that is, where the effect of price sensitivity is assumed to be the same for both brands. We then show how easy it is to relax this assumption and examine the consequences if in fact the PRICE effect differs by brand.

In this tutorial you will:

- Retrieve a previously saved model setup
- Re-estimate all models
- Determine the number of classes and name the segments
- Impose restrictions to simplify model
- Examine Output including share simulations for additional choice sets
- Include Brand x Price interactions in the model

LC Brand Pricing Example:

Participants in a choice-based conjoint study are administered six 3-alternative choice sets where each set poses a choice between alternative #1: Brand A (at a certain price), alternative #2: Brand B (at a certain price) and alternative #3: a None option. Brand A represents a new brand in this market. In total 7 distinct alternatives are defined in the file ABalt.sav:

	alt_id	brand	price	nobuy	price.a	price.b
1	A1	A	low	0	low	.
2	A2	A	medium	0	medium	.
3	A3	A	high	0	high	.
4	B1	B	low	0	.	low
5	B2	B	medium	0	.	medium
6	B3	B	high	0	.	high
7	No	.	.	1	.	.

Figure 1: Alternatives File

For our initial model, we will estimate effects for the attributes BRAND, PRICE and NOBUY. Later, we will replace the PRICE effect with the separate price effects PRICE.A and PRICE.B.

The six *active* choice sets are numbered 1,2,3,7,8 and 9 in the Sets File Sets.sav (see Figure 2 below). Additional sets, numbered 4,5,6,10,11, and 12, are called *inactive* sets (note that sets 10,11, and 12 contain just 2 alternatives). These inactive sets were *not* administered in the survey. Share estimates for both active as well as any inactive sets are generated automatically by Latent GOLD Choice, following estimation of a model. (This feature built into Latent GOLD replaces the need for a separate simulator program.)

	set_id	alt#1	alt#2	alt#3
1	1	A1	B1	No
2	2	A2	B1	No
3	3	A3	B1	No
4	4	A1	B2	No
5	5	A2	B2	No
6	6	A3	B2	No
7	7	A1	B3	No
8	8	A2	B3	No
9	9	A3	B3	No
10	10	A1	B3	.
11	11	A2	B3	.
12	12	A3	B3	.

Figure 2: Sets File

Response data on the 6 active sets that are used in this tutorial are not real, but were generated by John Wurst of SDR. The data were generated to reflect 3 market segments of equal size (500 cases for each segment) that differ on brand loyalty, price sensitivity and income. The *Loyal* segment has higher income and tends to be loyal to the existing brand B. The *Price Sensitive* segment has lower income, is *not* loyal but chooses solely on the basis of price. The 3rd segment, called *UpperMid*, contains primarily respondents with upper middle level income whose choice probabilities are somewhere between the other 2 segments.

The first 2 cases from the response data file BrandsAB.sav are shown in Figure 3 below. For each active set, the variable 'CHOICE' indicates whether the first, second or third alternative was selected.

	case_id	income	set_id	choice
1	1	lower	1	1
2	1	lower	2	2
3	1	lower	3	3
4	1	lower	7	1
5	1	lower	8	3
6	1	lower	9	1
7	2	upper middle	1	3
8	2	upper middle	2	3
9	2	upper middle	3	2
10	2	upper middle	7	3
11	2	upper middle	8	2
12	2	upper middle	9	1

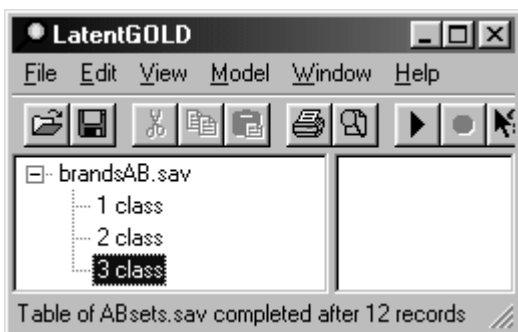
Figure 3: Responses to the active sets 1,2,3,7,8 and 9 for the first 2 cases.

To retrieve the previously saved model setup,

From the File Menu:

- Select Open
- Select BrandAB.lgf
- Click the Open button

The program retrieves the setups for the models named ‘1 class’, ‘2 class’ and ‘3 class’. Your screen displays the following:

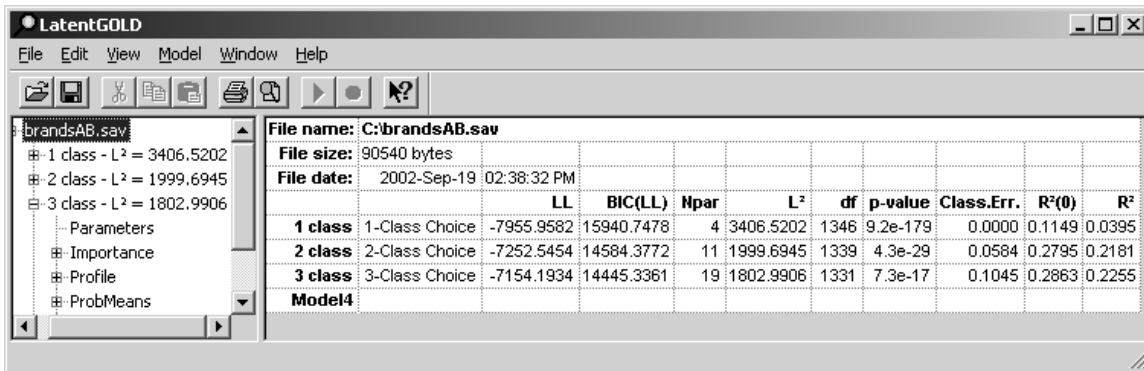


- Click on the file name ‘brandsAB.sav’ to highlight it
- Select ‘Estimate All’ from the Model Menu to estimate the 3 models

When estimation has completed

- Click on the file name 'brandsAB.sav' to highlight it

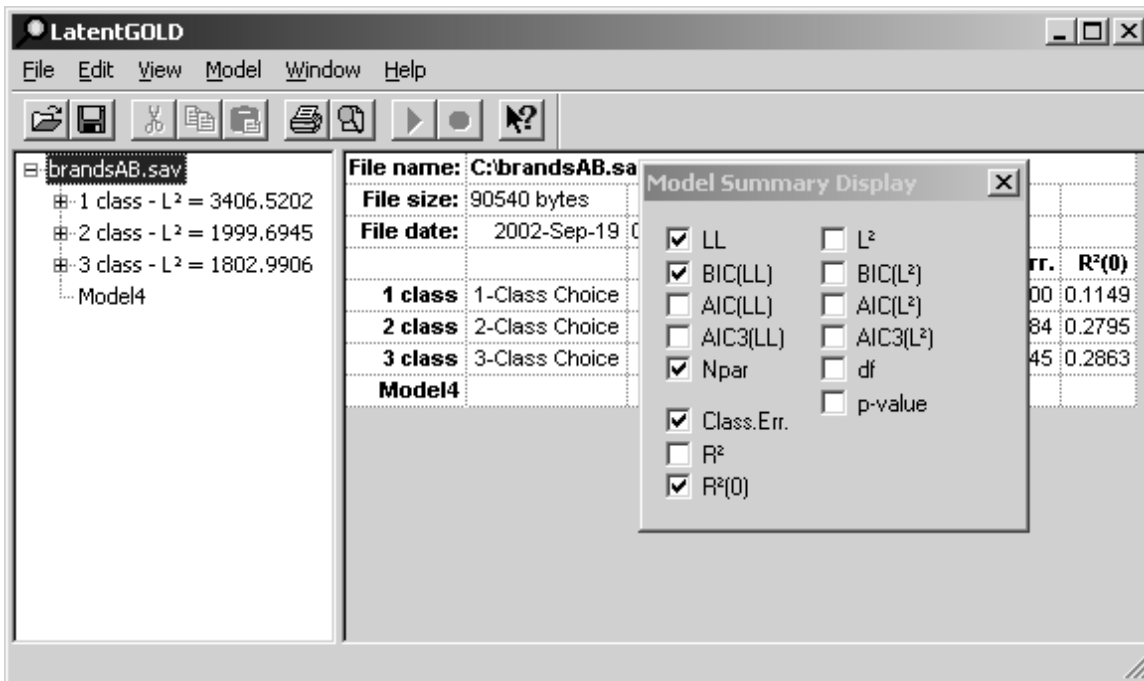
The Contents Pane displays the model summary output for the 3 models estimated



For this example we will utilize the bootstrap approach for p-value estimation, and we will use R²(0) instead of R² as a measure of model predictive performance¹. First, we will customize the model summary output to include only information relevant to this tutorial:

- Right click on the Contents Pane to retrieve the Model Summary Display Panel
- Remove certain check-marks (as shown below) by clicking the associated boxes.

The model summary output now contains a reduced number of columns:



¹The baseline model used to compute the R² statistic contains the alternative-specific constants (ASC), which are confounded with the brand and None effects here. That is, in this example each set always presents Brand A as the first alternative, Brand B the second, and None the third. Thus, to assess how well the model predicts the Brand and None effects, in addition to the Price effect, it is necessary to use the alternative *null* baseline model, which excludes the ASC, and predicts each alternative to be equally-likely. The performance of a model relative to this null model is assessed by R²(0).

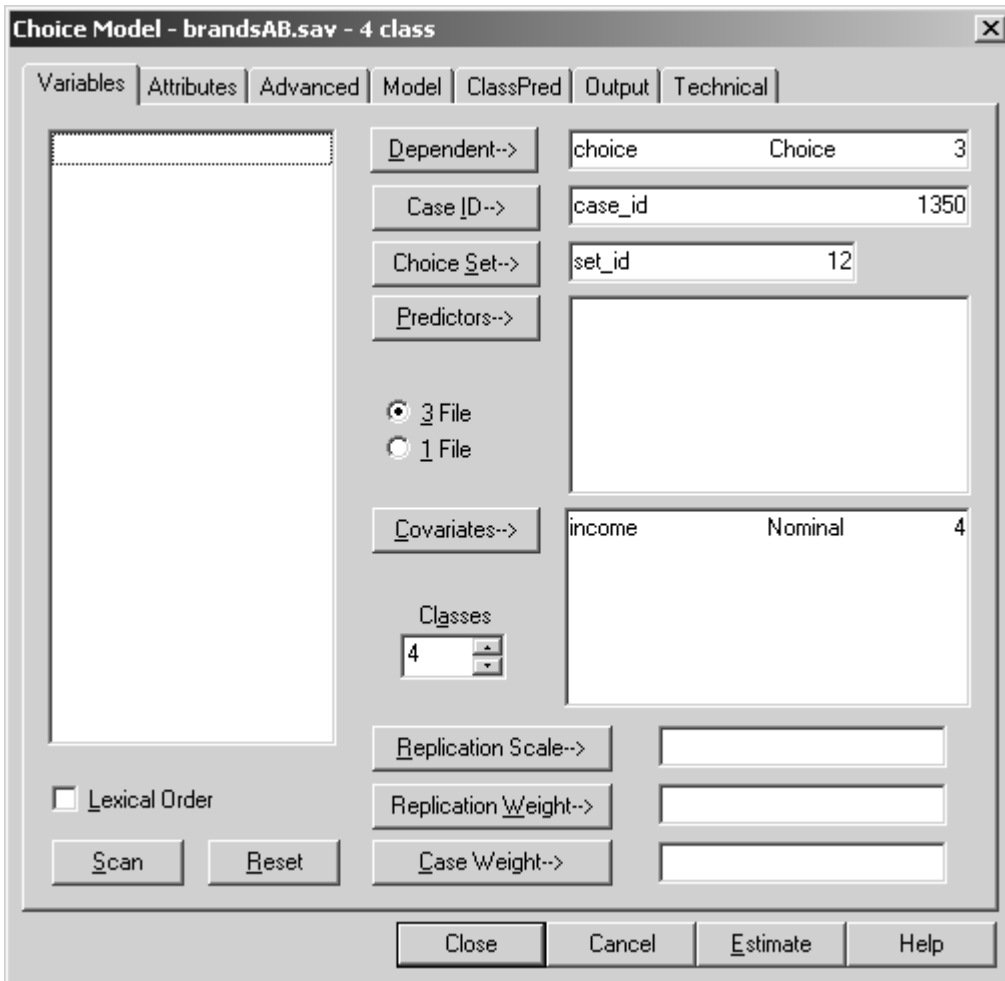
For comparison with the 3 models estimated, we will also estimate the 4-class model.

- Click once on the name 'Model4' to highlight it
- Click once (again) on the name 'Model4' to enter Edit mode
- Type the name '4 class' in place of 'Model4' and press Enter



To estimate the 4 class model,

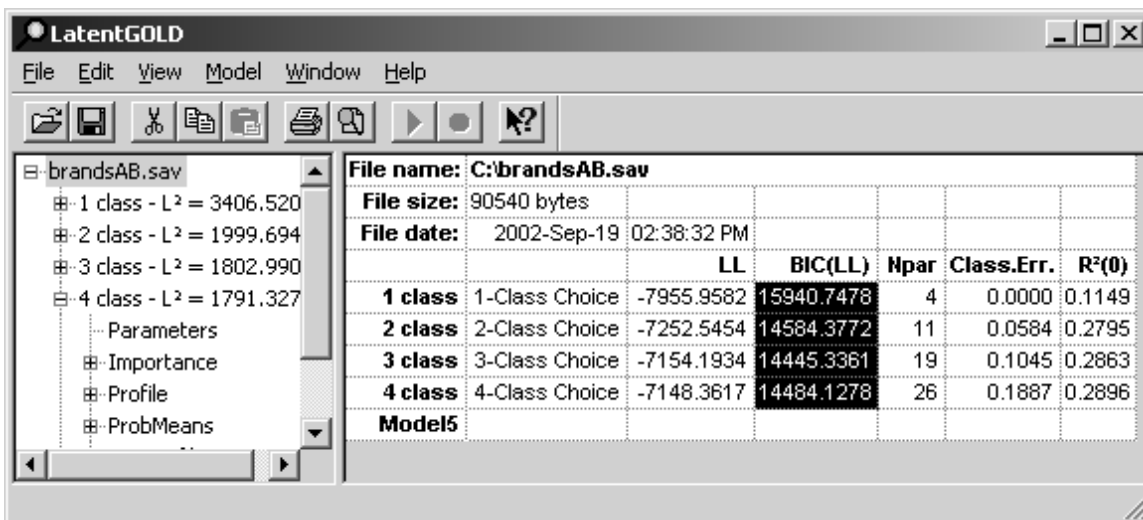
- Double click on the name '4 class' to open the model setup window
- Change the number of classes from '3' to '4'
- Click Estimate



After the estimation has completed,

- Click on the file name 'brandsAB.sav' to highlight it

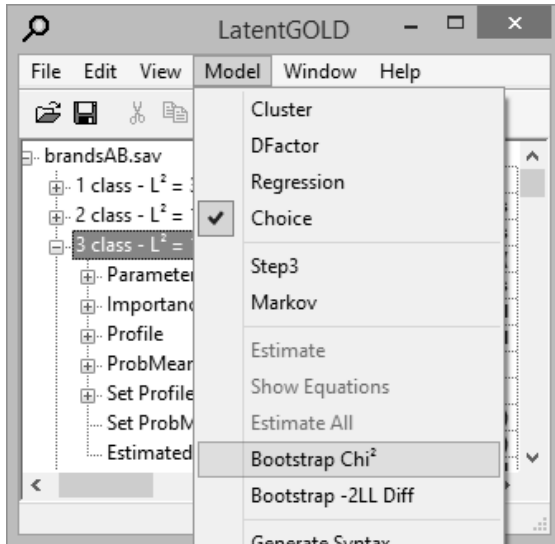
The Contents Pane now displays the model summary output for all 4 models estimated.



(Note: You may obtain a slightly different solution for the 4-class model -- one with $L^2 = 1784$ instead of $L^2 = 1791$. These solutions are very similar to each other and have a similar BIC value.)

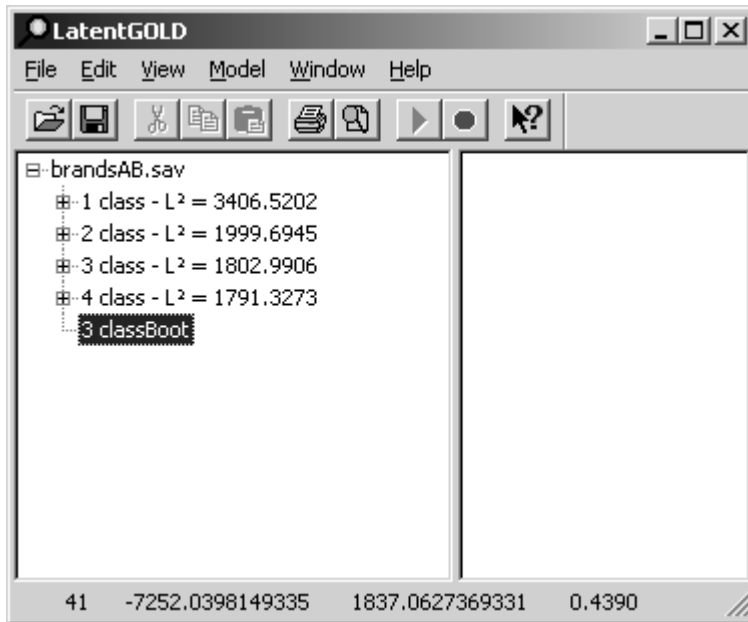
Notice that the BIC value is lowest for the 3-class model, correctly identifying the actual number of segments. In order to estimate the p-value for this model,

- Right click on '3-class' in the Outline Pane to retrieve the Model Menu (or, select '3 class' and then select 'Model' from the main menu) ➤ Select Bootstrap Chi²



The program creates a new model name, '3 classBoot', and begins to re-estimate the parameters of this model. After the estimation, it generates by default 500 random bootstrap samples based on these parameter estimates, one at a time, and computes the L^2 statistic for each. After each sample is processed, the program updates its then-current estimate for the p-value² displayed in the right-most portion of the status bar. The status bar at the bottom of the Figure below shows that after 41 bootstrap samples, the estimated p-value is .4390.

² The bootstrap estimate of the p-value is the proportion of bootstrap samples that yield an L^2 value higher than that obtained for the original model.



For this example, fewer than 500 samples are required to confirm (with 95% confidence) that the p-value is larger than .05. Thus, rather than wait for the completion of all 500 replications, we will terminate the current processing, reduce the number of samples from the default of 500 to 50, and re-estimate the model. For this,

- Click on the Red circular 'Cancel' icon
- Click Abandon
- Double-click on the 3-class Model and go to the Technical Tab
- Replace the number '500' with '50' Bootstrap L^2 replications
- Click Estimate

Now, to estimate the Bootstrap p-value,

- Right-click on the Model and select Bootstrap χ^2
- The Bootstrap p-value will now be estimated with only 50 replications

Upon completion, the Contents Pane contains the estimated p-value and its standard error -- .3400 and .0670 respectively, based on the 50 random bootstrap replications summarized below. From these results, we can be sure (with 95% confidence) that the p-value is in the acceptable range. The 95% confidence interval obtained using these bootstrap replications is $.3400 \pm 2(.0670)$.

Activated Constraints		1			
Random Seed		118536			
Best Start Seed		118536			
Bootstrap Seed		263463			
Chi-squared Statistics				Bootstrap	
Degrees of freedom (df)		1331	p-value	p-value	s.e.
L-squared (L ²)		1802.9906	7.3e-17	0.3400	0.0670

You may notice that the summary output for the 3-class model indicates 1 'Activated Constraint'. This is associated with the *descending* order restriction that was applied to the PRICE effect in the previously saved model that we used here. This restriction assures that for each segment the utility estimated for a *higher* price level is no larger than that estimated for a *lower* price level. The fact that this constraint was activated 1 time means that for one of the segments, there was a single instance where the unrestricted estimate would have violated this constraint. We will see where the constraint was activated as we now examine the Parameters Output from the 3-class model.

- Click on the + icon to the left of '3-class' to expand the output file list for this model
- Click on Parameters to display the part-worth utility parameter estimates

To examine the statistical significance of these estimates,

- Right click on the Contents Pane and select Z Statistic from the popup menu (or select Z Statistic from the View Menu)

Attributes	Class1	z-value	Class2	z-value	Class3	z-value	Mean S
A	-0.2866	-7.1602	-1.1525	-25.3457	0.0311	0.7360	-0.4692
B	0.2866	7.1602	1.1525	25.3457	-0.0311	-0.7360	0.4692
price							
low	0.4175	6.5759	0.0079	0.2377	1.2451	17.2713	0.5492
medium	0.0178	0.2566	0.0079	0.2377	0.0506	0.6703	0.0251
high	-0.4353	-6.4167	-0.0159	-0.2377	-1.2956	-13.9829	-0.5743
nobuy							
	0.0207	0.3561	-1.0411	-10.9498	0.6209	9.7157	-0.1354

Notice first that for Class 2, the estimate for the part-worth utility for both the *low* and *medium* price levels are identical (.0079). The exact equality of these estimates is due to the activation of the descending order restriction. We also see that the PRICE effects for this class is not significantly different from zero ($Z = .2377 < 2$). Without this restriction, these estimates would have exhibited a slight increase (-.0016 for the

low price and .0220 for the medium price due to sampling error (these numbers are obtained if the order restriction is removed). Use of the order restriction here eliminates this sampling error. Next, notice that the part-worth utility for brand B is largest (1.1525) for this class. Thus, we will name this class the 'Brand Loyal' segment.

The PRICE effect is largest for Class 3 (this is the 'Price Sensitive' segment), and the BRAND effect is not significant for this segment.

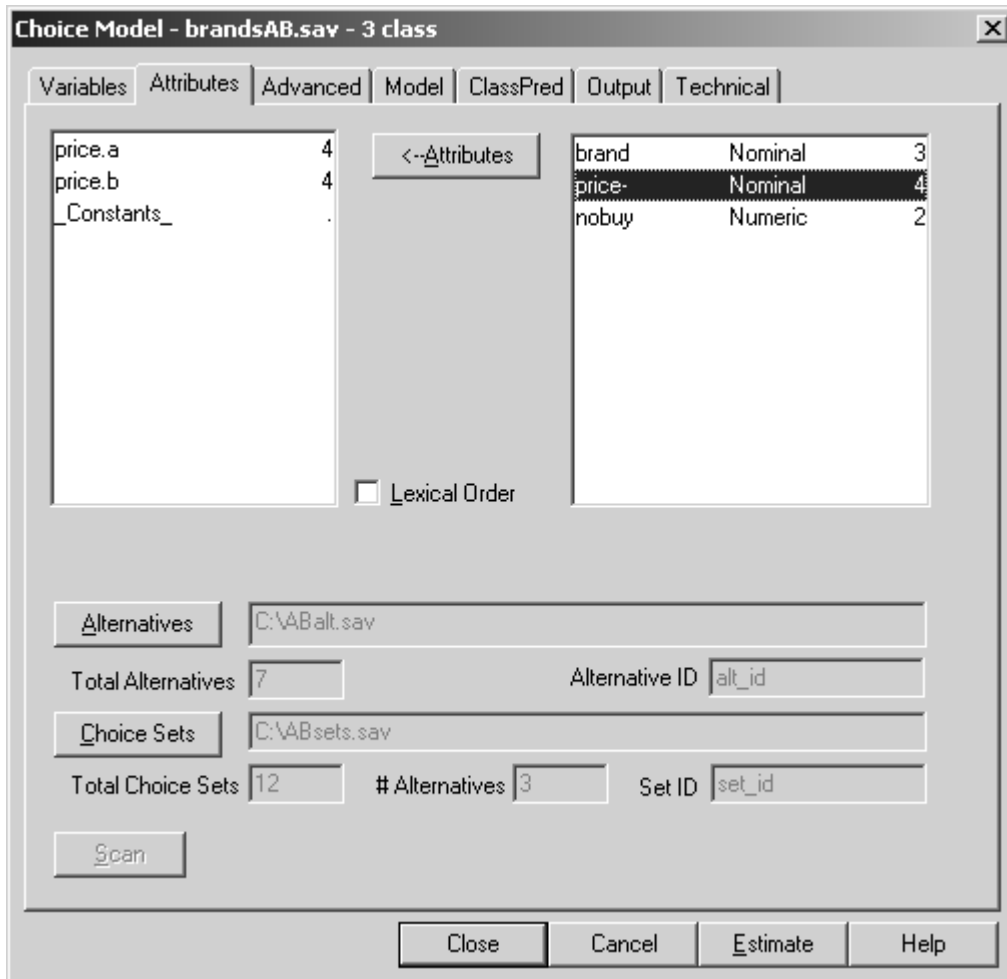
For class 1, both effects are significant, and like the Price Sensitive segment, the effect of the *medium* PRICE is seen to be approximately equidistant between the other 2 price levels. These results suggest that we can obtain a more parsimonious model by 1) changing the scale type of PRICE to numeric (with equidistant scores for the price levels) and 2) omitting the non-significant BRAND and PRICE effects from the model. To simplify the model in this way,

- Double click on 'Parameters' to open the model setup window
- Click on Model to open the Model tab
- Right Click on the PRICE cell for class 2 (in the column labeled '2') and select 'No effect' from the popup menu. The '2' in this cell changes to '-'.
- Right Click on the BRAND cell for class 3 (in the column marked '3') and select 'No effect' from the popup menu. The '3' in this cell becomes '-'.

Class	1	2	3	Class Independent	Order Restriction
brand (a)	1	2	-	No	None
price (a)	1	-	3	No	Descending
nobuy (a)	1	2	3	No	None

- Click Attributes to open the Attributes tab

Notice that the price attribute is displayed as 'PRICE-'. The appended '-' character indicates that the descending order restriction has been imposed.



To change the scale type for PRICE from Nominal to Numeric,

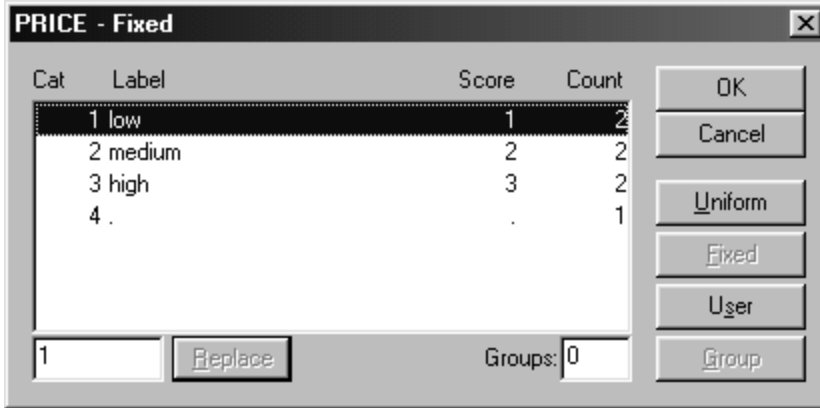
- Right click on PRICE- and select 'Numeric' from the popup menu

The label to the right of PRICE- changes from Nominal to Numeric.

To view the scores used to quantify the PRICE effect,

- Double click on PRICE-

The score window is now visible. The first column ('Cat') shows that among the 7 alternatives defined in the Alternatives file, PRICE contains 4 distinct categories. Column 4 ('Count') shows that 2 of these alternatives utilize the 'low' price which is assigned a score of '1', 2 utilize the 'medium' price scored '2', 2 others utilize the 'high' price scored 3, and the PRICE attribute is missing (Label = '.') for the remaining alternative (the 'None' option). These scores are appropriate, as the score assigned to the medium price ('2') is equidistant between the others.



To close the Score Window and continue with the estimation,

- Click OK
- Click Estimate
- Rename this model '3 class final'
- Click the data file name in the outline pane to display the model summary output

		LL	BIC(LL)	Npar	Class.Err.	R ² (0)
1 class	1-Class Choice	-7955.9582	15940.7478	4	0.0000	0.1149
2 class	2-Class Choice	-7252.5454	14584.3772	11	0.0584	0.2795
3 class	3-Class Choice	-7154.1934	14445.3361	19	0.1045	0.2863
4 class	4-Class Choice	-7148.3617	14484.1278	26	0.1887	0.2896
3 class	3-Class Choice	-7154.1934	14445.3361	19	0.1045	0.2863
3 classBoot	3-Class Choice	-7154.1934	14445.3361	19	0.1045	0.2863
3 class final	3-Class Choice	-7155.0441	14418.2061	15	0.0972	0.2849

Note that the BIC for the 3-class final model is the lowest. In the right-most column we see that the change in R²(0) shows improved prediction from about .11 to .28 as we moved from the aggregate (1-class) model to the 3-class final model.

To further examine the predictive performance,

- click on '3 class final'
- scroll down to the prediction statistics

We see that over all 6 sets, this model correctly predicts 471 of the 1,663 alternative #1 (Brand A) choices, 3,759 of the 4,173 alternative #2 (Brand B) choices, and 867 of the 2,264 alternative #3 (None) choices. This corresponds to a hit rate of 63% $(471 + 3,759 + 867) / 8,100 = .63$, for a prediction error of .37 (= 1 - .63).

LatentGOLD

File Edit View Model Window Help

brandsAB.sav

1 class - L² = 3
 2 class - L² = 1
 3 class - L² = 1
 4 class - L² = 1
 3 class - L² = 1
 3 classBoot - L² =
 3 class final - L² =
 Parameters
 Importance
 Profile
 ProbMeans
 Set Profile
 Set ProbMeans
 Model18

Prediction Statistics

Error Type	Baseline(0)	Baseline	Model	R ² (0)
Squared Error	0.6667	0.6143	0.4767	0.2849
Minus Log-likelihood	1.0986	1.0230	0.8159	0.2573
Absolute Error	1.3333	1.2275	0.9811	0.2642
Prediction Error	0.6667	0.4829	0.3707	0.4439

Prediction Table

Observed	Estimated			Total
	1	2	3	
1	471.0	749.0	443.0	1663.0
2	99.0	3759.0	315.0	4173.0
3	251.0	1146.0	867.0	2264.0
Total	821.0	5654.0	1625.0	8100.0

Files

To examine the predictive performance separately for each class and separately within each set,

- Click 'Set Profile'

LatentGOLD

File Edit View Model Window Help

brandsAB.sav

1 class - L² = 3406
 2 class - L² = 1999
 3 class - L² = 1802
 4 class - L² = 1791
 3 class - L² = 1802
 3 classBoot - L² =
 3 class final - L² =
 Parameters
 Importance
 Profile
 ProbMeans
 Set Profile
 Set ProbMeans
 Model18

	Class1	Class2	Class3	Overall	Observed	StdResid
Class Size	0.3376	0.3297	0.3327			
Set Average (n=8100)						
Choice 1	0.2480	0.0826	0.2778	0.2034	0.2053	0.3736
2	0.4263	0.8248	0.3044	0.5171	0.5152	-0.2343
3	0.3257	0.0926	0.4178	0.2795	0.2795	-0.0000
Set 1 (n=1350)						
Choice 1	0.2754	0.0826	0.3943	0.2514	0.2585	0.5185
2	0.4811	0.8248	0.3943	0.5655	0.5148	-2.4755
3	0.2435	0.0926	0.2114	0.1831	0.2267	3.7429
Set 2 (n=1350)						
Choice 1	0.1991	0.0826	0.1606	0.1479	0.1252	-2.1690
2	0.5317	0.8248	0.5465	0.6332	0.6326	-0.0292
3	0.2692	0.0926	0.2930	0.2189	0.2422	1.8326
Set 3 (n=1350)						
Choice 1	0.1399	0.0826	0.0532	0.0922	0.0630	-3.5335
2	0.5710	0.8248	0.6164	0.6698	0.6837	0.6258
3	0.2891	0.0926	0.3304	0.2381	0.2533	1.1487

At the top of this output file, we see that respondents are equally divided among the 3 classes (Class Size = .3376, .3297 and .3327). Next, we see that among all 8,100 choices, the Brand Loyal segment (class 2), selects the Brand B alternative 82.48% of the time.

The 'Overall' column shows that overall, the model predictions closely approximate the observed choices. The standardized residuals in the right-most column are all within the expected range (-2,+2), which show that the difference between these predictions and the observed proportions fall within the normal sampling variation.

As the price charged for Brand A increases as we move from Set 1, to Set 2 to Set 3, we see a substantial loss in share among the price sensitive segment (class 3). When the price for both Brand A and Brand B are low (Set 1), about 39% of this segment choose Brand A. When the price increases to the medium range (Set 2), the percentage choosing Brand A declines to 16%, and when the price becomes high (Set 3), only 5% choose the Brand A alternative.

The standardized residuals indicate that the within-set predictions fall outside the normal bounds that would be expected if the model were true.

To see how the results change if we include a Brand X Price interaction in the model.

- Double click on the new Model at the bottom of the Contents Pane
- Click on Attributes to Open the Attributes tab
- Click on PRICE- and click the Attributes button to remove the linear PRICE effect from the model
- Highlight PRICE.A and PRICE.B and click the Attributes button to enter these interactions in the model
- Right click and select 'Numeric' to change the scale type from Nominal to Numeric
- Click Estimate



Upon completion of the estimation, click the data file name BrandsAB.sav to display the model summary screen.

			LL	BIC(LL)	Npar	Class.Err.	R ² (0)
bandsAB.sav							
1 class - L ² = 3406.5	1 class	1-Class Choice	-7955.9582	15940.7478	4	0.0000	0.1149
2 class - L ² = 1999.6	2 class	2-Class Choice	-7252.5454	14584.3772	11	0.0584	0.2795
3 class - L ² = 1802.9	3 class	3-Class Choice	-7154.1934	14445.3361	19	0.1045	0.2863
4 class - L ² = 1791.3	4 class	4-Class Choice	-7148.3617	14484.1278	26	0.1887	0.2896
3 class - L ² = 1802.9	3 class	3-Class Choice	-7154.1934	14445.3361	19	0.1045	0.2863
3 classBoot - L ² = 18	3 classBoot	3-Class Choice	-7154.1934	14445.3361	19	0.1045	0.2863
3 class final - L ² = 18	3 class final	3-Class Choice	-7155.0441	14418.2061	15	0.0972	0.2849
	Model6	3-Class Choice	-7143.8045	14424.5583	19	0.0894	0.2843

Note that the BIC for our 3-class final model is still the best. To save the 3 class final model for future use,

- click '3 class final' in the outline pane

- from the File menu select 'Save Definition'
- Click the Save button