

Design and Analysis of Choice Experiments Using R: A Brief Introduction

Hideo Aizaki*¹⁾ and Kazushi Nishimura²⁾

1) National Institute for Rural Engineering, National Agriculture and Food Research Organization, 2-1-6 Kamondai, Tsukuba-city, Ibaraki, 305-8609, Japan

2) National Agricultural Research Center for Kyushu Okinawa Region, National Agriculture and Food Research Organization, 2421 Suya, Koshi-city, Kumamoto, 861-1192, Japan

Abstract

This paper briefly introduces choice experiments using R, which is a language and environment for statistical computing. Choice experiments belong to the family of stated preference methods and are applied to numerous issues in research fields such as marketing science, transportation economics, environmental economics, agricultural economics, or health economics. We explain the process of creating choice sets using the AlgDesign package and the process of statistically analyzing responses using the survival package. Since R is free software, readers of this document may find it easy to apply choice experiments to their research projects.

Key words

choice-based conjoint analysis, design of experiments, factorial design, discrete choice model, conditional logit model, survival package, AlgDesign package

Introduction

This paper, which is a briefly modified English edition of Aizaki and Nishimura (2007), aims to explain the design and analysis of choice experiments using R (R Development Core Team 2006). Choice experiments, also called as choice-based conjoint analysis, choice modeling, or stated choice method, belong to the family of stated preference methods and are applied to numerous issues in research fields such as marketing science, transportation economics, environmental economics, agricultural economics, or health economics (see Louviere *et al.* 2000, Ryan and Farrar 2000, Bennett and Blamey 2001, Bateman *et al.* 2002, Hensher *et al.* 2005). Choice experiments comprise seven steps: characterizing the decision process, identifying and describing the attributes, developing an experimental design, developing the questionnaire, collecting data, estimating model, and interpreting the results of policy analysis or decision support (Champ *et al.* 2003). However, this paper only describes a method for develop-

ing an experimental design (designing choice sets) and estimating model (statistically analyzing responses) using R. Detailed descriptions of the other steps and practical techniques for choice experiments are referred to in the abovementioned books or papers. R and its packages that are used in this paper are supposed to be downloaded from the official web site of R (The R Project for Statistical Computing <<http://www.r-project.org/>>) and installed onto your personal computer. Readers who are not familiar with R should refer to web sites, books, or documents related to R (a list of publications is displayed on the web site).

Creating choice sets using the AlgDesign package

Attributes, levels, and question

Following the hypothetical example of consumers' valuation of milk, which is based on previous research (Sawada *et al.* 2002, Iwamoto *et al.* 2003), we explain the method for designing the choice sets and statistically analyzing the response data for choice experiments using R. Fig. 1 shows an example of a question for choice experiments, wherein respondents are assumed to be asked to select one among three alternatives, including two choices of

* Corresponding Author
E-mail: aizaki@affrc.go.jp

Q1. Please circle the item listed below that you are most likely to purchase.

1. Milk A 2. Milk B 3. Neither milk A nor milk B is preferred

	Milk A	Milk B
HACCP label	No	Yes
Eco label	No	No
Price per liter	145 yen	160 yen

Fig. 1 An example of choice experiment questions

Table 1 Total choice sets excluding the none-of-these option

Question	Milk A			Milk B		
	HACCP	Eco	Price	HACCP	Eco	Price
1	No	No	145 yen	Yes	No	160 yen
2	Yes	No	155 yen	Yes	Yes	145 yen
3	Yes	Yes	150 yen	No	Yes	160 yen
4	No	No	150 yen	No	No	145 yen
5	No	Yes	155 yen	Yes	Yes	150 yen
6	Yes	Yes	145 yen	Yes	No	155 yen
7	No	Yes	160 yen	No	Yes	155 yen
8	Yes	No	160 yen	No	No	150 yen

milk and none of the two (i.e., neither milk A nor milk B is preferred). Each type of milk is expressed by three attributes: “HACCP label,” “Eco label,” and “Price per liter.” The HACCP (hazard analysis of critical control points) label indicates that the HACCP system has been used for food safety control during the process of manufacturing milk in a factory. The Eco label shows that the raw milk is produced by cows that are fed by ecological dairy farmers who manage their dairy farms in harmony with the environment. Each of these attributes has two levels: “Yes” or “No.” The price per liter attribute comprises four levels: 145 yen, 150 yen, 155 yen, and 160 yen (\$1=approximately 115 yen as of October 2007). Each type of milk is supposed to be the same except for these three attributes.

Respondents are assumed to be asked to answer eight questions, as shown in Fig. 1. Table 1 shows total choice sets, and each line describes the two types of milk (alternatives). For example, line 1 in Table 1 corresponds to question 1 (Fig. 1), and it shows that the respondents can select one among two types of milk in question 1—milk A is characterized by costing 145 yen without both the HACCP and Eco labels and milk B is characterized by costing 160 yen with the HACCP label and without the Eco label.

How to create choice sets

Following the first of four methods explained by Louviere *et al.* (2000, p. 114), the total choice sets can be created through five steps using R (Fig. 2).

Step 1: The function `gen.factorial()` included in the AlgDesign package (Wheeler 2004a) is used for creating a full factorial design.

Step 2: The function `optFederov()` included in the AlgDesign package (Wheeler 2004b) is used for generating a fractional factorial design ((mixed) orthogonal array) from the full factorial design.

Step 3: Making M-1 copies of the fractional factorial design.

Step 4: In order to create a choice set with M alternatives, randomly select one of the alternatives (rows) from each of the M sets of the fractional factorial design without replacement. Repeat this step until all alternatives in each of the M sets of the fractional factorial design are assigned to P choice sets.

Step 5: Translating the design codes in the choice sets into codes with a unique and corresponding level, and then adding the none-of-these option to each choice set optionally.

Step 1: Creating a full factorial design

A full factorial design with two two-level factors (HACCP label and Eco label) and one four-level factor (price attribute) can be created using the function `gen.factorial()`. In the full factorial design, two or more attributes (independent variables) are manipulated and all combinations of the levels of each attributes are included. In our case, where there are two two-level attributes and one four-level attribute, the full factorial design comprises sixteen combinations of the levels of each attributes (=2×2×4). After attaching the AlgDesign package by using the function `library()`, the full factorial design is generated and assigned to an object `ffd` using the function `gen.factorial()` as follows.

```
> library(AlgDesign)
> ffd <- gen.factorial(c(2,2,4), varNames=c("HAC",
"ECO", "PRI"), factors="all")
> ffd
      HAC  ECO  PRI
1      1    1    1
2      2    1    1
3      1    2    1
4      2    2    1
5      1    1    2
6      2    1    2
7      1    2    2
8      2    2    2
9      1    1    3
10     2    1    3
11     1    2    3
12     2    2    3
13     1    1    4
14     2    1    4
15     1    2    4
16     2    2    4
```

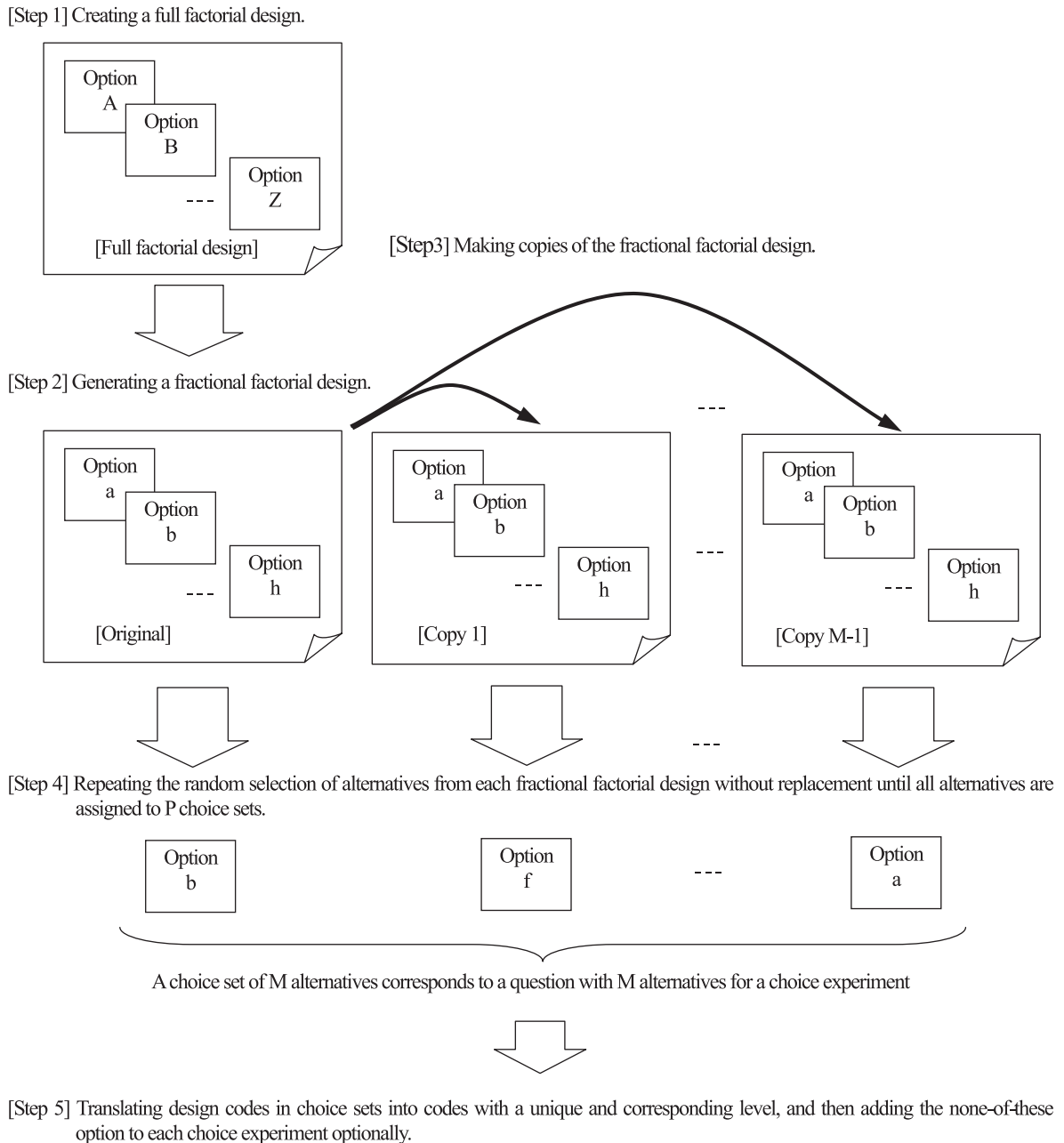


Fig. 2 Steps followed in constructing choice sets

The function `gen.factorial()` has three arguments: `c()`, `varNames`, and `factors`. The argument `c()` in the function is used for setting the attributes and levels included in a full factorial design. In order to construct a full factorial design having two two-level attributes and one four-level attribute, the argument `c()` is assigned values of 2, 2, and 4. If a full factorial design with three two-level attributes, two four-level attributes, and one six-level attribute needs to be created, the argument `c()` will be “`c(2,2,2,4,4,6)`.” The argument `varNames` is used for assigning the names of the attributes to each column in the full factorial design. Here, the first, second, and third columns are called `HAC` (HACCP label attribute with two levels), `ECO` (Eco label attribute

with two levels), and `PRICE` (price attribute with four levels), respectively. The last argument `factors="all"` indicates that all attributes are factors. There is normally no need to modify this argument as long as the steps described in this paper are followed.

Executing the abovementioned command creates the full factorial design with two two-level attributes and one four-level attribute, and the design is assigned to the object `ffd`. The name of the object (`ffd`), which can be arbitrarily decided, lists the elements of the object. We find that the full factorial design comprises sixteen rows.

Step 2: Generating a fractional factorial design

Consider that we have to create a half-fractional factorial design where the number of rows of the full factorial design is reduced from sixteen to eight. The function `optFederov()` generates the fractional factorial design from the full factorial design, which is then assigned to an object `des`, after arbitrarily setting the random seed by using the function `set.seed()` as follows.

```
> set.seed(54321)
> des <- optFederov(~., ffd, 8)
> des
$D
[1] 0.25
$A
[1] 6.333333
$Ge
[1] 1
$Dea
[1] 1
$design
   HAC  ECO  PRI
1    1    1    1
4    2    2    1
5    1    1    2
8    2    2    2
10   2    1    3
11   1    2    3
14   2    1    4
15   1    2    4
$rows
[1] 1 4 5 8 10 11 14 15
```

The function `optFederov()` has three arguments: `~.`, `ffd`, and `8`. The argument `~.` implies that all data variables are used linearly and their names are used in a model. The argument `ffd` indicates the name of data containing the candidate list, which is the same as the name of the object containing the full factorial design created above. The argument `8` indicates the number of rows (alternatives) in the fractional factorial design.

The object `des` contains some objects (elements) as a result of the function `optFederov()`. The fractional factorial design that is required below and is called `design` in the object `des` is assigned to an object `alt1` from the object `des` by a notation `$`, which is used to access an element of the object, as follows.

```
> alt1 <- des$design
> alt1
   HAC  ECO  PRI
1    1    1    1
4    2    2    1
```

5	1	1	2
8	2	2	2
10	2	1	3
11	1	2	3
14	2	1	4
15	1	2	4

Step 3: Making copies of the fractional factorial design

M-1 copies of the fractional factorial design created in step 2 are made. Since two packs of milk are shown for each question in the choice experiments (M=2), a copy of the fractional factorial design must be made. The copy is created in an object `alt2` using the operator `<-` in the following manner.

```
> alt2 <- alt1
> alt2
   HAC  ECO  PRI
1    1    1    1
4    2    2    1
5    1    1    2
8    2    2    2
10   2    1    3
11   1    2    3
14   2    1    4
15   1    2    4
```

If two or more copies of the fractional factorial design are required, the names of the objects containing these copies must be different from each other.

Step 4: Creating choice sets by using random selection without replacement

We randomly select rows (alternatives) from each of the M fractional factorial designs without replacement, and this selection is repeated until all rows in each of the M sets of the fractional factorial design are assigned to P total choice sets. P is equal to the number of rows in the designs. In R, a different and new uniform random variable is added to every fractional factorial design; then, each design is sorted on the basis of its corresponding random variable.

First, a new uniform random number is added to the object `alt1` as follows.

```
> alt1 <- transform(alt1, r1=runif(8))
> alt1
   HAC  ECO  PRI    r1
1    1    1    1  0.05374091
4    2    2    1  0.75569340
5    1    1    2  0.15476521
8    2    2    2  0.12866907
```

```
10  2  1  3  0.11170085
11  1  2  3  0.23430698
14  2  1  4  0.91222889
15  1  2  4  0.78156383
```

The command `transform(alt1, r1=runif(8))` implies that a uniform random variable `r1` containing eight values is added to the object `alt1`. The number of values contained in the random variable is equal to the number of rows in the object `alt1`. If the object `alt1` comprises `n` rows, the argument `runif(8)` is changed to `runif(n)`. Similarly, the following command is implemented in order to add another uniform random variable `r2` to the object `alt2`.

```
> alt2 <- transform(alt2, r2=runif(8))
> alt2
      HAC  ECO  PRI      r2
1     1    1    1  0.55358492
4     2    2    1  0.17772093
5     1    1    2  0.97593965
8     2    2    2  0.68168019
10    2    1    3  0.72350066
11    1    2    3  0.81742403
14    2    1    4  0.02218425
15    1    2    4  0.49822342
```

If the number of fractional factorial designs is `M`, the abovementioned command is repeated `M` times.

Next, each fractional factorial design is sorted on the basis of its corresponding uniform random variable. The following command is executed on the object `alt1` by using the function `order()`.

```
> alt1_sort <- alt1[order(alt1$r1),]
> alt1_sort
      HAC  ECO  PRI      r1
1     1    1    1  0.05374091
10    2    1    3  0.11170085
8     2    2    2  0.12866907
5     1    1    2  0.15476521
11    1    2    3  0.23430698
4     2    2    1  0.75569340
15    1    2    4  0.78156383
14    2    1    4  0.91222889
```

A similar command is executed on all fractional factorial designs (only `alt2` in our case).

```
> alt2_sort <- alt2[order(alt2$r2),]
> alt2_sort
```

```
      HAC  ECO  PRI      r2
14    2    1    4  0.02218425
4     2    2    1  0.17772093
15    1    2    4  0.49822342
1     1    1    1  0.55358492
8     2    2    2  0.68168019
10    2    1    3  0.72350066
11    1    2    3  0.81742403
5     1    1    2  0.97593965
```

Each line of the sorted fractional factorial designs (objects `alt1_sort` and `alt2_sort`) corresponds to each question of the choice experiments (the upper side of Fig. 3). For example, the same line of the objects `alt1_sort` and `alt2_sort` shows the combination of the attributes levels of milk A and milk B for the same question, respectively. The first line of the object `alt1_sort` shows the combination of the attributes levels of milk A for question 1, whereas the eighth line of the object corresponds to the same for question 8. Similarly, the first line of the object `alt2_sort` shows the combination of the attributes levels of milk B for question 1, while the eighth line corresponds to the same for question 8. When two or more alternatives having the same combination of attributes levels appear in one question, it is necessary to repeat step 4.

Step 5: Translating the design codes

In step 5, let us translate each design code in the choice sets into each unique and corresponding level (lower side of Fig. 3). Following the relationship between the design codes and levels shown in Table 2, fractional factorial designs shown in the upper side of Fig. 3 are translated to the designs shown in the lower side of Fig. 3 that are equivalent to Table 1.

Analyzing responses using the survival package

Representative component of utility

A conditional logit model, which has been applied in numerous previous researches related to choice experiments, is assumed to be used for the statistical analysis of the responses to the choice experiments questions in our example. This model is based on the random utility theory where a respondent's utility is divided into two components: a representative (systematic) component and a random component. The representative component of utility for the none-of-these option is normalized to be 0 and that for the two types of milk is assumed to be as follows.

$$V_j = ASC + \beta_H HAC_j + \beta_E ECO_j + \beta_P PRI_j \quad (1)$$

where V_j denotes the representative component of utility for milk j ($j=A, B$). ASC is the alternative specific constant. HAC_j takes a

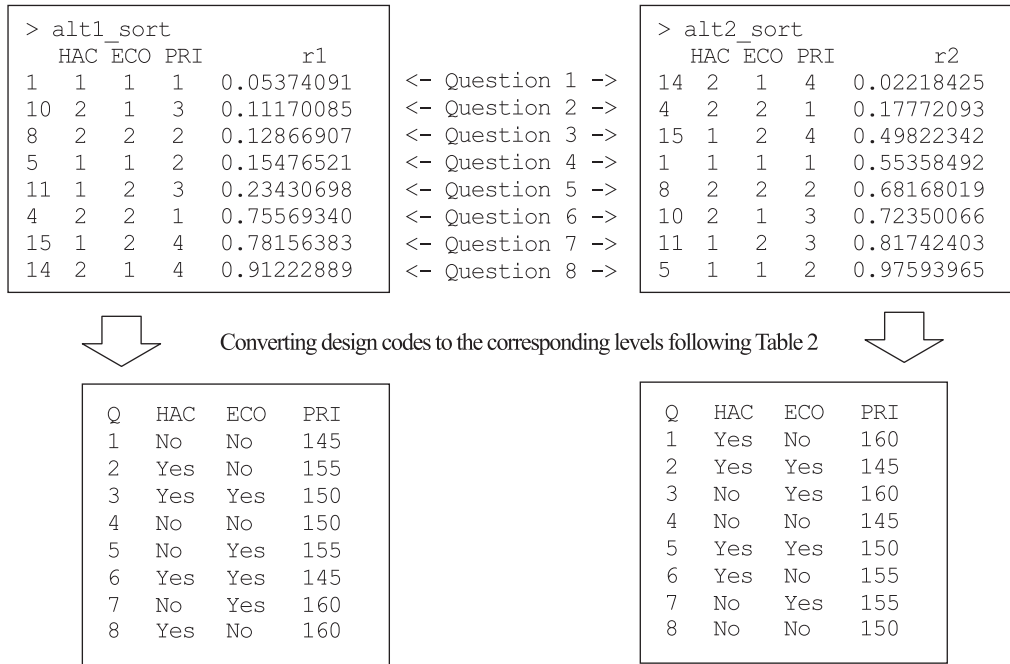


Fig. 3 Creating total choice sets (excluding the none-of-these option) from sorted fractional factorial designs

Table 2 Relationship between the design codes and levels in each attribute

Attribute	Design codes			
	1	2	3	4
HAC	No	Yes		
ECO	No	Yes		
PRI	145 yen	150 yen	155 yen	160 yen

value of 1 if milk j was produced by using the HACCP system and 0 otherwise. ECO_j takes a value of 1 if milk j was obtained from ecological dairy farmers and 0 otherwise. PRI_j denotes the price of milk j . β_H , β_E , and β_P are unknown parameters associated with HAC_j , ECO_j , and PRI_j , respectively.

The conditional logit model can be executed by using the function `clogit()` that is included in the survival package (Lumley 2006). Frequently, in order to consider the effects of individual characteristics on the valuation of attributes, the interaction between individual characteristics and attribute variables are included in the model. The following two cases are shown: first, independent variables comprising only attribute variables and second, independent variables comprising attribute variables and individual characteristics.

Creating and reading data set

Suppose that ten consumers responded to the eight questions for the choice experiments (Table 1). For the purpose of simplicity, the sample size is set to be smaller than that in the case of practical choice experiments. The responses data listed in Table 3 have been randomly created.

Table 3 10 individuals' responses to the eight questions for the choice experiments

Respondent	Question ^a							
	1	2	3	4	5	6	7	8
1	3	2	2	1	3	3	1	3
2	3	2	1	3	2	1	3	1
3	1	2	3	1	1	1	2	1
4	3	3	3	1	2	1	3	3
5	3	2	1	1	3	1	3	3
6	1	3	3	3	3	1	3	3
7	3	3	3	1	2	1	3	3
8	2	2	3	2	2	3	3	1
9	3	2	2	2	2	1	3	2
10	3	1	3	3	2	2	3	3

^a A value of 1 implies selecting milk A, 2 implies selecting milk B, and 3 implies selecting the none-of-these option.

Fig. 4 shows a part of a data set for the function `clogit()`, which is created from Table 1 and Table 3. Each respondent answers eight questions and each question comprises three alternatives, that is, milk A, milk B, and the none-of-these option. Each alternative comprises one row of the data set. Therefore, the data set comprises 240 rows, with 10 blocks (10 individuals) having eight sub-blocks (eight questions) and three rows (three alternatives). For example, the data for respondent 1 are listed from row 2 to row 25, where rows 2, 3, and 4 denote data related to milk A, milk B, and the none-of-these options, respectively, in question 1.

Variable *STR* in column A is a stratification variable that is used to identify each combination of respondents and questions. The variable is assumed to be as follows.

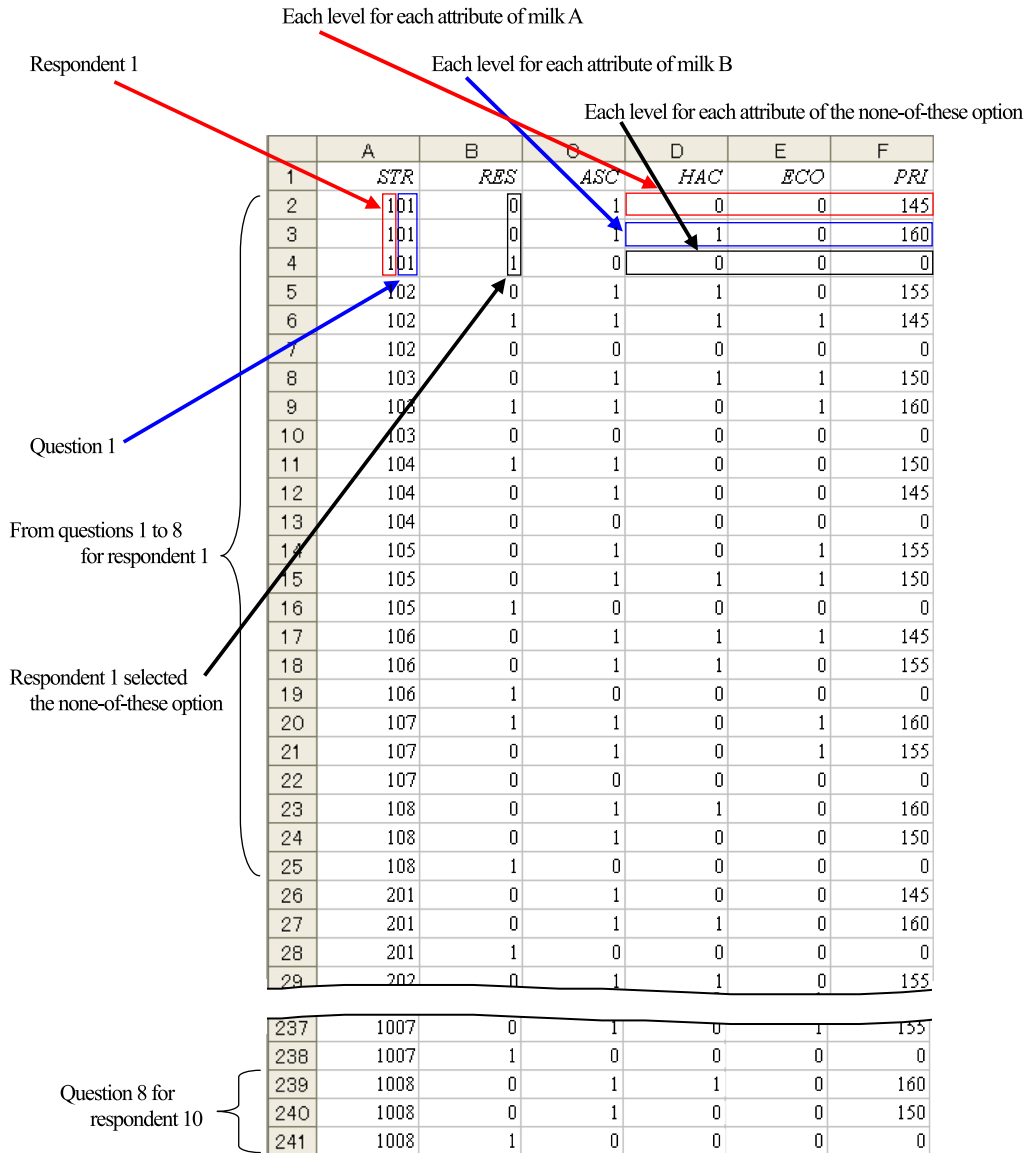


Fig. 4 A part of a data set for the function `clogit()`

$$STR = 100 \times N + Q \quad (2)$$

where variable $N (=1, 2, 3, \dots, n)$ is the respondent's identification number and variable $Q (=1, 2, 3, \dots, q)$ is a question number. Since there are three alternatives to each question (choice situation), the stratification variable takes a different value every three lines in our case. For example, the value of STR in the case of question 1 for respondent 1 from row 2 to row 4 is 101 ($=100 \times 1 + 1$), while the value of STR in the case of question 8 for respondent 10 from row 239 to row 241 is 1008 ($=100 \times 10 + 8$).

Variable RES in column B is a dummy variable that, at each row, takes the value 1 if the alternative is selected and 0 otherwise. For example, we observe that respondent 1 selected the none-of-these option (third alternative) in the case of question 1 because the variable RES in row 4 takes a value of 1 and that in both rows 2 and 3 takes a value of 0.

Columns C to F contain independent variables in the conditional logit model. Variable ASC in column C is an alternative specific constant for each type of milk, A and B. Variable HAC in column D and variable ECO in column E represent dummy variables for the HACCP label and Eco label, respectively. Each dummy variable takes a value of 1 if the milk has each label and 0 otherwise. Variable PRI in column F denotes the price of milk. Each attribute variable of the HACCP label, Eco label, and price corresponding to the none-of-these option takes a value of 0. Because the function `clogit()` does not correspond to a question (choice scenario) format including the none-of-these option, it is necessary to normalize the representative component of utility for the option to be 0.

Suppose the data set file "data1.txt" is saved as a tab delimited file in the directly "c:/test." The command `read.delim()` can be used to read the data set file and create a data set object called

data1 in R as follows.

```
> data1 <- read.delim("c:/test/data1.txt")
```

Applying the conditional logit model

The conditional logit model can be applied by using the function `clogit()` included in the survival package in R. After attaching the survival package, the function `clogit()` is executed as follows.

```
> library(survival)
> clogout1 <-
clogit(RES~ASC+HAC+ECO+PRI+strata(STR), data=data1)
> clogout1
Call:
clogit(RES~ASC+HAC+ECO+PRI+strata(STR), data=data1)
      coef  exp(coef)  se(coef)    z    p
ASC  12.6659  3.17e+05  5.2041  2.434  0.0150
HAC   0.8378  2.31e+00  0.4087  2.050  0.0400
ECO   0.3972  1.49e+00  0.4308  0.922  0.3600
PRI  -0.0918  9.12e-01  0.0342 -2.686  0.0072
Likelihood ratio test=22.9 on 4 df, p=0.00013 n=240
```

The function `clogit()` has two arguments — the formula for the estimation model and the name of the data set. The formula is assumed to be “RES~ASC+HAC+ECO+PRI+strata(STR),” where the left-hand side of the operator~shows the response variable (RES) and the right-hand side shows independent variables (ASC, HAC, ECO, and PRI) and the stratification variable (strata(STR)). This model assumes that the representative component of utility for milk is to be a linear combination of independent variables (see equation (1)). The data set argument is data1. The output from the function `clogit()` is assigned to the object `clogout1` that includes some other objects (elements). For example, the output of the function `clogit()` by default does not exhibit a log-likelihood at zero and at convergence. These values are listed by executing the following command.

```
> clogout1$loglik
[1] -87.88898 -76.41796
```

The output shows that the values of the log-likelihood at zero and that at convergence are -87.88898 and -76.41796, respectively.

Considering the effects of individual characteristics on their valuation

A method for reflecting the respondents’ individual characteristics in the conditional logit model is to use a new variable that multiplies the individual characteristic variable and attribute vari-

Table 4 10 individuals’ responses to eight questions for the choice experiments and their value for the variable FEM

Respondent	Question ^a								FEM ^b
	1	2	3	4	5	6	7	8	
1	3	2	2	1	3	3	1	3	1
2	3	2	1	3	2	1	3	1	0
3	3	2	3	1	1	3	3	3	1
4	3	3	3	1	2	1	3	3	0
5	3	3	3	3	3	1	3	3	1
6	1	3	3	3	3	1	3	3	0
7	3	3	3	3	2	1	3	3	1
8	2	2	3	2	2	3	3	1	0
9	3	3	2	2	2	1	3	2	1
10	3	1	3	3	2	2	3	3	0

^a A value of 1 implies selecting milk A, 2 implies selecting milk B, and 3 implies selecting the none-of-these alternative.

^b A value of 1 implies the respondent is a female and 0 otherwise.

ables (including ASC). Suppose that the effect of the respondent’s gender on their valuation of the price variable is statistically tested, that is, a new variable $PRI_j FEM$ is introduced as one of independent variables, where variable FEM takes a value of 1 if the respondent is a female and 0 otherwise. The representative component of utility for milk considering the effect of the variable FEM on the variable PRI_j is as follows.

$$V_j = ASC + \beta_H HAC_j + \beta_E ECO_j + \beta_P PRI_j + \beta_{PF} PRI_j FEM \tag{3}$$

where β_{PF} is the coefficient for the variable $PRI_j FEM$. An interaction between two variables in formula is expressed by using the operation :, that is, PRI:FEM. A data set (data2) for this model is made from Table 1 and Table 4 that has been randomly created. The conditional logit model can be estimated as follows.

```
> clogout2 <-
clogit(RES~ASC+HAC+ECO+PRI+PRI:FEM+strata(STR), data=data2)
> clogout2
Call:
clogit(RES ~ ASC + HAC + ECO + PRI + PRI:FEM +
strata(STR), data = data2)
      coef  exp(coef)  se(coef)    z    p
ASC  9.81506  1.83e+04  5.68911  1.725  0.084
HAC  0.84498  2.33e+00  0.43903  1.925  0.054
ECO  0.47206  1.60e+00  0.45666  1.034  0.300
PRI -0.07533  9.27e-01  0.03725 -2.022  0.043
PRI:FEM
      -0.00217  9.98e-01  0.00311 -0.697  0.490
Likelihood ratio test=34.5 on 5 df, p=1.87e-06 n= 240
```

The coefficient of the variable `PRI:FEM` has a negative sign but

it is not significant ($p=0.490$), implying that the estimated coefficient of the variable `PRI` for females is not statistically different from that for males.

Conclusion

This paper briefly illustrates a method for designing choice sets and statistically analyzing the responses for choice experiments using R. Readers who are interested in both choice experiments and R can attempt other methods of design and analysis using R.

References

- Aizaki, H. and K. Nishimura (2007) Introduction to Choice Experiments Using R (Dēta-Kaiseki-Kankyō R ni yoru Sentakugata-Konjōinto-Bunseki Nyūmon), Technical Report of the National Institute for Rural Engineering (Nouso Kōgaku Kenkyū-Syō Gihō), 206: 151–173 (in Japanese with English Summary), URL <http://nkk.naro.affrc.go.jp/library/publication/seika/giho/206/206-11.pdf>.
- Bateman, I., R. T. Carson, B. Day, M. Hanemann, N. Hanley, T. Hett, M. Jones-Lee, G. Loomes, S. Mourato, E. Ozdemiroglu, D. W. Pearce OBE, R. Sugden and J. Swanson (2002) Economic Valuation with Stated Preference Techniques: A Manual, Edward Elgar, Cheltenham, UK.
- Bennett, J. and R. Blamey (Eds.) (2001) The Choice Modelling Approach to Environmental Valuation, Edward Elgar, Cheltenham, UK.
- Champ, P. A., K. J. Boyle and T. C. Brown (2003) A Primer on Nonmarket Valuation, Kluwer Academic Publishers, Dordrecht, Netherlands.
- Hensher, D. A., J. M. Rose and W. H. Greene (2005) Applied Choice Analysis, Cambridge University Press, Cambridge, UK.
- Iwamoto, H., Y. Yamamoto, K. Sato, and M. Sawada (2003) Effects of HACCP and Eco Labels on Japanese Consumers' Choice of Milk, Contributed paper presented for the 47th Annual Conference of the Australian Agricultural and Resource Economics Society, Fremantle, The Esplanade Hotel, February 11–14, <<http://www.aares.info/files/AARES/rest/Iwamoto.pdf>>, browsed on 29 November 2007.
- Louvière, J. J., D. A. Hensher and J. D. Swait (2000) Stated Choice Methods, Cambridge University Press, Cambridge, UK.
- Lumley, T. (2006) `clogit`. S original by T. Therneau, ported by T. Lumley: The survival package version 2.29.
- R Development Core Team (2006) R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0. URL <http://www.r-project.org/>.
- Ryan, M. and S. Farrar (2000) Using Conjoint Analysis to Elicit Preferences for Health Care, *BMJ: British Medical Journal*, 320: 1530–1533.
- Sawada, M., H. Iwamoto and Y. Yamamoto (2002) Consumers' Valuation of HACCP and ECO Labeled Milk (Syōhisya no HACCP oyo-bi Kankyō-Taisaku-Ninshō-Hyōji Gyuunyū ni taisuru Kachi-Hyōka), in Higuchi, A. and K. Awaji (Eds.), Changes in Agricultural Environment and Policy (Nōgyō no Yōken-Hendō to Taiō-Saku), Association of Agricultural and Forestry Statistics, Tokyo, Japan, 46–58 (in Japanese).
- Wheeler, R. E. (2004a) `gen.factorial`. AlgDesign. The R project for statistical computing. URL <http://www.r-project.org/>
- Wheeler, R. E. (2004b) `optFederov`. AlgDesign. The R project for statistical computing. URL <http://www.r-project.org/>

Received December 3, 2007

Accepted March 20, 2008