

HIERARCHICAL LOGIT MODELS:
A NEW SOFTWARE PACKAGE

N Smith
Research Fellow

F Milthorpe
Systems Analyst
Transport Research Centre
Macquarie University
New South Wales

ABSTRACT

Many transport demand problems involve discrete choices such as travel mode choice, choice of destination, type and number of automobiles. In some cases these choices can be satisfactorily described by a multinomial logit model but in other cases a nested structure is required to avoid model restrictions. This paper presents a new, extremely user friendly computer package for both micro-computers and mainframes which uses a full information maximum likelihood technique (FIML) to jointly estimate hierarchical logit models. It also estimates single stage multinomial logit and ordinary least squares regressions quickly and efficiently.

INTRODUCTION

The demand for transport services is an important field for the application of computer modelling techniques. Some of the decisions which define the demand for transport can be measured by continuous variables such as kilometres travelled or litres of fuel used, enabling the application of conventional linear regression estimates to establish the link between the choice and the variable factors which might influence the choice. However many more of the transport demand decisions involve discrete choices such as route, type and number of automobiles and the one so often quoted in the discrete demand literature because it makes such an excellent example, travel mode. Unfortunately the discrete dependent variable introduces a non-linearity into an otherwise linear regression model. The multinomial logit model is a relatively low cost and effective way of resolving these difficulties but should only be applied if the variance between any pair of alternatives is independent of which pair is being compared. This restriction known as the "Independence of Irrelevant Alternatives" (IIA), see Hensher and Johnson (1981) is likely to be violated in many travel choice situations, for instance the choice between the three modes of car, Ansett, or Australian airlines. Mc Fadden(1977) proposed nested logit models to apply in such cases. The choice set is divided into a hierarchy of subsets of alternatives and the choice modelled within and between subsets. These estimations may be made sequentially but we will contend that they are better estimated simultaneously. In addition choice based sample weighting allows us to study problems where a random sample of users choices is not practical... If our choice set contains an infrequent choice(e.g bicycle to work), we would need an unnecessarily large number of observations of the most popular choice to obtain a sufficient number of such responses in a true random sample.

Many computer modelling programs have been developed and applied to these various situations both for main frame computers and for personal computers. But the practitioner faced with establishing a satisfactory explanation for his own particular real world situation can have problems choosing the appropriate method. Of course in the airline choice example the last 2 choices are very similar, but not all cases of IIA violation are so glaring. Again he may intend to use a regression method to model kilometres travelled, only to find that the data might be better considered in a series of ranges. To begin again with a new model in a new package is often not practical, instead time and money constraints make it necessary to accept a less-than-preferred solution.

This paper explains a new, very user friendly, computer package, HLOGIT(Hierarchical Logit), which allows the user to apply modeling techniques of increasing sophistication to demand data without having to set up the problem again and again. Its simplicity means that the facility to simultaneously estimate nested logit models will be available to many more people. We have written it with Axel Boersch-Supan from Mannheim University, West Germany. It is based on the Macquarie University MNL program BLOGIT (Hensher and Johnson (1981))and on Boersch-Supan's private program used in his Ph.D thesis, Boersch-Supan (1984) and his subsequent work in housing demand modelling Boersch-Supan (1985). The new program is not a simple combination of elements of these programs; it is new program with major advances in the interface with input data, specification of models for estimation and available output. It has always been

difficult to formulate discrete choice problems because the user needs to specify a lot of information. We believe that this process has been made as painless as possible in a method which has the computer think like a person rather than requiring the user to think like a computer.

The paper is organised as follows. We begin by briefly outlining the underlying econometric theory behind the techniques encompassed in the package. The second part discusses the computational practice, while the third part explains in detail the simple requirements for setting up models and running the program. We conclude with an illustrative example of three stage hierarchical logit model estimation.

THEORETICAL BACKGROUND

Consider a person t , who chooses one of M discrete alternatives (e.g. a commuter choosing a mode of travel to work from car, bus, train). Each alternative i , $i = 1, \dots, M$, has an associated indirect utility (measure of desirability) for that person, u_{it} . This will be dependent on a set of x_{it} alternative specific attributes (e.g. fare, travel time) and a set of y_t personal characteristics (e.g. income) plus

some additive random component ε_{it} which captures unobserved variables. Thus we say that

$$u_{it} = v_{it}(x_{it}, y_t; \beta) + \varepsilon_{it} \quad (1)$$

where u_{it} is the indirect utility function with parameters β dependent on the variables but independent of the individual and v_{it} is the observed component of the indirect utility. The unobserved or random utility components have a joint distribution function $F(\varepsilon_{1t}, \dots, \varepsilon_{Mt})$ for all t . Person t will choose alternative i if the indirect utility derived from it is greater than that derived from each and every other alternative in the choice set

$$u_{it} > u_{jt} \quad j \neq i \quad \forall j \in M \quad (2)$$

Since the random component is unobserved, we attach a probability to an event

$$P_{it} = \text{Prob}[u_{it} > u_{jt}]$$

or

$$P_{it} = \text{Prob}[v_{it} - v_{jt} > \varepsilon_{jt} - \varepsilon_{it}] \quad (3)$$

To operationalise this general specification requires a specific assumption about the distribution of the ε , $F(\varepsilon)$.

Multinomial Logit [MNL]

If we assume that for each alternative the random component ϵ is equally and independently distributed with an extreme value type I distribution we can write $F(\epsilon)$ as

$$F(\epsilon_t) = \exp [- \exp(\epsilon_{1t}) - \exp(\epsilon_{2t}) - \dots - \exp(\epsilon_{Mt})] \quad (4)$$

then the probability of person t choosing alternative i is

$$P_{it} = \exp(v_{it}) / \sum \exp(v_{jt}) \quad (5)$$

Equation (5) is the basic multinomial logit model MNL (see Hensher and Johnson (1981)) for a complete derivation of the model).

The ratio P_{it} / P_{jt} depends only on the difference $(v_{it} - v_{jt})$ regardless of any other alternatives. This independence of irrelevant alternatives (IIA), property is a major restriction of multinomial logit. The problems posed by this restriction can be illustrated by the infamous red bus / blue bus paradox. A group of commuters original choice for travel is car or a bus painted red. Another bus, identical in all respects except its colour, blue, is introduced to the route. Sense dictates that this bus would simply split the existing bus market; however an MNL model would also switch travellers from cars, in order to maintain the ratio of the probabilities of selecting the car and the red bus.

Heirarchical Logit [NMNL]

Hierarchical logit or Nested Multinomial Logit overcomes this problem by partitioning the larger choice sets into subsets of fewer similar alternatives, and models the choice as first between subsets, then within each subset. The choice between subsets is based on aggregate utility indices of the subset of alternatives. Weighting each of the aggregate indices overcomes the IIA problem. The MNL form of the distribution $F(\epsilon)$, equation (5), is rewritten as

$$F(\epsilon) = \exp \left[- \sum_{j=1}^N \left(\sum_{i=1}^{M_j} (\epsilon_{it})^{1/d_j} \right)^{d_j} \right] \quad (6)$$

Equation (6) is factorised into N subsets each of which contains M_j alternatives (McFadden, 1978, 1979). This process can be reiterated to generate multistage hierarchical choice models (McFadden, 1983).

The taste weights for the aggregate utility indices are called Dissimilarity parameters because a higher value of d_j implies less correlation of the unobserved attributes ϵ_{jt} in subset j . For $d_j = 1$ the NMNL model collapses to MNL.

Choice Based Sampling

To derive precise results for all alternatives in a random sample, the sample size depends on the most infrequent choice. However if the actual proportion of occurrence of each alternative in the total population is known population weights can be used to adjust over-sampled infrequent choices and under-sampled frequent choices. The estimator proposed by Manski and Lerman (1977), the weighted ML - WESML; reweights the observations inversely to the sampling ratios i.e. the proportion of alternative i in the population divided by the proportion of alternative i in the sample.

COMPUTATIONAL PRACTICE

The product of the choice probabilities for all T observations in the data set is termed the likelihood of the set. The best estimates of the model's coefficients are those values which maximize the likelihood of observing the choices made by each person t in the set of T . When the natural log of the likelihood is taken we have

$$\text{Log } L_{\text{MNL}} = \sum_{t=1}^T \sum_{i \in (1..M)} s_{it} \log [P(i | x_i, \beta)]$$

for $t = 1, \dots, T$ observations and $i = 1, \dots, M$ alternatives. $s_{it} = 1$ if person t selected alternative i and zero otherwise, with the β s the parameters on the explanatory variables x_i . The package currently accommodates hierarchical models of up to 3 levels so the log likelihood expression associated with the Full Information Maximum Likelihood estimates has the form

$$\text{Log } L_{\text{FMNL}} = \sum_{t=1}^T \sum_{i \in (1..M)} s_{it} \log [P(i | x_i, \beta, \theta, \tau)]$$

where the θ 's are the dissimilarity parameters at the second level and the τ 's the dissimilarity parameters for the upper levels. Simultaneous estimation of this equation yields statistically efficient estimates.

This method is preferable to Sequential-Nested Logit which estimates the model sequentially by applying MNL to each subset, constructing aggregate indirect utilities, then applying MNL between subsets. That procedure is computationally simpler than FIML but has 3 problems: second stage standard errors are incorrect, parameter restrictions cannot be enforced between subsets and there is substantial efficiency loss for $M > 3$. Amemiya (1981) has proposed a correction for upper level standard errors in sequential models to ensure asymptotically efficient parameter estimates but the calculations involved in such a correction of a sequential estimate make the total procedure just as complex as a FIML estimation.

Optimal FIML values are obtained using a modified hill climbing procedure which equalises the influence of the utility parameters β and the dissimilarity parameters $d(\theta, \tau)$. This is a fast optimization procedure, written specifically for HLOGIT. It is based on the method reported by Goldfeld and Quandt (1972) but is not the computer optimization routine GQOPT. It takes that step at each iteration that maximizes a quadratic approximation to the function on a sphere of suitable radius. First and second analytical derivatives of the log likelihood function are calculated. The analytic second derivatives are used as the default but faster numeric derivatives are available as an option. This method was chosen instead of an algorithm relying on first derivatives, such as BHHH, because approximation of the Hessian tends to be unsatisfactory given the highly non-linear behaviour of the dissimilarity parameters.

Whereas MNL model has a proven unique optimum, guaranteeing fully efficient parameter estimates, multiple local optima can exist for FIML-NL (McFadden (1981), Cosslett (1978)). Hence it is important that "good" starting values be selected. Having a package capable of both MNL and FIML-NL means that starting values may be simply obtained by an initial MNL estimation as suggested by Hensher (1986).

The user has to satisfy two simple requirements. He needs a set of data containing choice information and some variables which affect the choice. This data can be in a wide variety of forms limited only by the need to be on an observation by observation basis, as distinct from variable by variable. Then a set of commands must be created which (1) describe the data set to the program, (2) perhaps ask for some statistics, and (3) request models to be estimated. Normally the first commands describing the data set will remain unchanged, thus these are entered from a file. The statistics and model commands can also be entered from a file; then the user will simply edit the model request section to invoke different modelling strategies, but in these sections the option exists to run

interactively. A major advantage of HLOGIT lies in the the common set up commands which mean extra of model types can be invoked with minimal effort. The following, not exhaustive, list of HLOGIT commands is presented, to demonstrate the ease of setting up models in this program.

DATA COMMANDS FOR HLOGIT

Here we seek to explain the commands in general terms. Full documentation is available in the program users manual. All main commands start in column 1, and stay in effect until the program finds another command. Other lines are subject only to the restriction that they do not start in column 1. The length of commands is in general limited by the commonsense of the user rather than constraints of the program. No other package places less constraints on its user. Commands are entered in free format, lower-case/upper-case letters are not distinguished except in the case of user defined names. If you enter COST, Cost or cost the program keeps it in that form enabling COST and cost to be two different variable names if required.

Defining Choices- ALTS

Somewhere in each observation a choice must be registered. This is usually by a number. This command identifies these with an up to 8 character name. For example

ALTS 1 = train 3 = car 4 = airplane

These alternatives need not be ordered, they simply name existing numbers. If instead the choice is registered by a set of variables corresponding to each alternative, set to one if the alternative is chosen and zero otherwise, the command becomes

1 = train = modetrn 2 = car = modecar etc.

here 1 and 2 are numbers assigned by the user and modetrn and modecar being the names given to the presence/ absence variables in the data set. The final permissible form of registering choice is via an alpha character thus

1 = train = 'T' 2 = car = 'C' etc

but alpha characters can only appear in a formatted data set.

Identifying elements in the data set - READ

This command enables the package to read in the data. It lists the names of the variables in their order of appearance in the data set. As with ALT names, the name must begin with alpha character, up to 7 characters may follow, in any combination of alpha, numeric or the special characters " \$ % @ & ". There is no preset expected variable order and

only a few simple conventions apply. If the choice variable has not been specifically named in the ALT command it must be named ALT. Alternative specific variables have the appropriate alternative attached in brackets, the bracketed name being the first 3 letters of the alternative name for example, cost(tra). The data are generally grouped by either variable :

cost(tra),cost(car), cost(bus), time(tra),time(car),time(bus)

or by alternative :

cost(tra),time(tra), cost(car),time(car), cost(bus),time(bus)

In either case the cross hatch abbreviation makes this specification much simpler, especially for long variable lists. The first example above then becomes

cost(#),#=#tra,car,bus , time(#),#=#tra,car,bus

and the second

(cost(#),time(#),#=#tra,car,bus)

A set of subcommands then order the way the data set is to be handled. Three such subcommands are

- (1) Format : followed by either a fortan style format statement for the data or either of the words "free" or "binary"
- (2) Nobs = N1 - N2 , where N1 is the number of the first observation required, N2 that of the last. This allows for the choice of subsets of data.
- (3) Skip = N, skips over N physical records at the top of the data set to skip headings such as "BLOGGS TRAVEL DATA"

Personal Characteristics for different alternatives - ASSIGN

Personal characteristics, although their values are independent of alternatives, sometimes need to be assigned to one or more alternatives in order to act as shift effects with respect to that particular alternative. For instance an increase in the size the family travelling party could be expected to have a greater influence on choice of public transport modes than it would on the choice of car. The ASSIGN command allows this distinction to be made:

ASSIGN nfam alts = air,tra name = nfampub
 nfam alt = car name = nfamcar

Data transformation - COMPUTE

It is often necessary to recode variables or create new variables. The package provides its own intelligent equation reader to perform arithmetic, functional and logical transformations. # allows for the short hand expression of multiple transformations. Again only a few rules apply, each compute statement must begin on a new line but can continue over multiple lines. The following example shows some of the different forms

```
COMPUTE
  tohrs(#) = inveht(#) + waitmin(##)/60
  income = income / 1000
  dist = log (dist)
  if nfam == 1 or nfam == 2 then small = 1 else small = 0
```

Subsets of Data - RANGE

The command RANGE obs = N1 - N2 or
if (income > 20000)

The first option restricts the following procedures to a subset of observations by number. The second range option 'if' selects the cases for which the if statement is true, for example where income exceeds \$20,000.

Data Descriptive Commands

A useful variety of data descriptive commands are implemented since these are often needed in conjunction with choice modelling.

LIST (var1, var2) allows particular variables to be examined and in general would be used in conjunction with a range restriction.

FREQ (var) gives frequencies of desired variable(s) or can be combined with subcommands to break the frequency tables down by alternative or produce cross tabulations.

MEAN (var1,var2.....) encompasses a set of descriptive statistics: mean, standard deviation, variance, minimum, maximum and median, by alternative if required by a subcommand. The covariance and correlation matrices are subcommand options.

HIST (var1, var2) - provides histograms of the named variables

PLOT x = var1 y = var2 provides a two-way plot

SETTING UP MODELS IN HLOGIT

The current version of the HLOGIT package has three model commands.

Regression Models - OLS

Running ordinary least squares regression models requires only subcommands identifying dependent and independent variables. A constant is included by default unless the user specifies "noconst". For example

```
OLS      dep = income
         indep = nfam, ncars, work
         noconst
```

The Multinomial Logit command - MNL

Since most of the specification usually needed in setting up MNL models has already been completed, a single command will now run a logit model. For example

```
MNL      indep = cost, time, income, nfampub, nfamcar
```

If starting values are to be set or constraints imposed on the parameter estimates, they can be entered in the indep command, either as (starting value) or (= constraint), or (> / <constraint)

```
MNL      indep = cost ( 0.5), time ( > 0 < 1)
```

Here β_{cost} has a starting value of 0.5 and the constraint $0 < \beta_{\text{time}} < 1$ applies.

The full choice set is assumed unless a subcommand limits alternatives. The Alts subcommand is also needed when population weights are to be specified. For example

```
alts = tra (0.5), car ( 1.2), air ( .7)
```

where the bracketed factor is the proportion of the alternative in the sample divided by the known proportion of the alternative in the population.

The Nested Multinomial Logit Model - NMNL

The nested multinomial logit command can contain all the subcommands of the MNL model but follows the `alt` command, if present, with an essential `tree` command which describes the hierarchical grouping of the alternatives using a bracketing technique. Consider a commuter's choice of travel mode from the choice set train, bus, car driver, car passenger, walk and bicycle. We might decide to break down this choice into a hierarchy where the first choice is longer distance modes [train, bus, car driver or passenger] or short distance modes [walk or bicycle]. The long distance modes could then be further divided into public or private modes. Tree diagrams are commonly drawn to represent such decisions. We use a bracketing technique to enter the "tree" into the computer, writing

$$\text{tree} = ((\text{tra}, \text{bus}), (\text{crd}, \text{crp})), (\text{wal}, \text{bik})$$

Having chosen this tree we now want to name its elements so that the package can provide meaningful output. Continuing with the "tree" nomenclature we refer to the upper level decision as the limb, moving down through the branch to the elemental alternative twigs.

```
branch = (tra,bus) = public
branch = (crd,crp) = private
branch = (wal,bik) = short
limb = (pub,pri) = long
limb = (sho) = short
```

The dissimilarity parameters (θ s and τ s) which we have to estimate in addition to the β s in NMNL models use these names and hence are given their starting values or constraints in these commands

```
branch = (tra, bus) = public (1.0)   This starts  $\theta_{\text{pub}}$  at 1.0
branch = (tra, bus) = public (=0.5)  This sets  $\theta_{\text{pub}}$  at 0.5
```

The `indep` command specifying the model takes the same form as in the MNL model but because starting values are much more important in NMNL they are often drawn via subcommand from stored values from a previous model calculation. This option is available in MNL and in the branch/limb commands, but is the usual option in NMNL.

Changing Optimization Criteria - OPTIMIZER

This optional command in the model section allows the various step size, maximum number of iterations and convergence criteria in the optimizer to be changed. A full "pull down" menu assists the user in this process. If the program is running interactively the

menu is just amended by the user. If it is running from a set up file it looks for a previously prepared optimizer menu. If none is present it creates such a file then terminates, waiting to be rerun after the user amends the optimizer menu. This menu also allows the user to specify numeric rather than, the default, analytic second derivatives.

Commands for Running the Package

A set of file handling and recall commands allow data sets to be stored and recalled at different stages of the program. Likewise output can be spooled to files. Of particular interest to users with large data sets is the provision for buffering data into the program. HLOGIT does not limit the the number of observations. However if the data exceeds the size of the internal work sheet, data has to be moved between a scratch file and memory ("paging") which slows the program down substantially. The worksheet in the standard version of the program accommodates 16000 numbers. The number of observations (MAXOBS), accomodated in a problem can be calculated, from the number of alternatives (NALTS), the number of alternative specific attributes (NATTS), and the number of personal characteristics(NCHAR), via the formula:

$$\text{MAXOBS} = 16000 / (\text{NALT} + \text{NALT} * \text{NATTS} + \text{NCHAR})$$

Output Considerations - OUTPUT

The use of names rather than numbers for alternatives and variables means the output is easily read. The three- character abbreviations for alternatives revert to their eight-character names, "train" not "tra". This is the second reason for the use of names. The main reason is that we believe it makes the modelling process much less prone to error. It is much easier to confuse V5 and V6 than "income" and "sex". Of course a contrary user can still name his variables X1,X2.....

Trees can be output drawn in the conventional form to be certain the specified tree structure is correct. It is wise to invoke this output option, *Tree*, when preparing a new complex model.

Output commands govern both the type and quantity of output, from output at each step to simply final values. The usual log likelihood values , parameter estimates together with t - statistics and standard errors are provided for the model . There are additional types of output available by subcommand. For example *Goodness* provides goodness of fit tests. The Lagrange Multiplier and Wald tests are asymptotic tests based on the maximum likelihood method, see Engle (1983). *Elast* produces selected elasticities, aggregate, disaggregate, direct and cross. *Predic* writes tables of prediction success: (1) Observed shares and frequencies against predicted shares and frequencies and (2) Percentage correctly predicted , and utility of predicted choices.

The outputs of the package are continuously being updated in line with users needs and suggestions. For instance we have a contour plotting option in preparation.

AN EXAMPLE

The illustrative example refers to one of a set of experimental data sets prepared for testing HLOGIT. It is presented to demonstrate the advantages of using Full Information Maximum Likelihood estimation on a data set where choices can be hierarchically structured. The results are illustrative only.

Figure 1. shows the HLOGIT instructions for reading in the data set, then running an MNL model to model the mode choice of 118 travellers who had travelled between major capital cities. These people were travelling for private reasons on vacation, visiting friends or relatives so were not subject to the constraints of the business traveller. Their choice was between the four currently available modes of intercity travel, airplane, coach(bus), train or car. This example has the advantage of capturing the complete choice set, since walk is not a viable alternative and all modes were available to all travellers.

Figure 1: HLOGIT Setup for Data Input and Running MNL

```
TITLE Intercity Travel Mode Choice

ALTS 1 = airplane
      2 = train
      3 = bus
      4 = car

READ alt (cost(#),time(#),termtime(#),frequent(#),wait(#))
      #=air,tra,bus,car, hinc,npers,sex,age
      Format (f1.0, 4(3f5.0,f2.0,f3.0), f5.0,f2.0)

ASSIGN hinc alts=air name=hinc_air
        npers alts=air name=npers_air

COMPUTE time(#) = time(#)/60 #=air,tra,bus,car
         if ( alt eq 4 ) frequent(#) = 1
         else frequent(#) = frequent(#)/18 #=air,tra,bus,car

MNL
  indep = cost,termtime, hinc_air, npers_air, frequent, time
```

The set of mode specific explanatory variables used in the model are Cost, the cost of travel by the chosen mode, Termtime, the time spent waiting at terminals, Frequency the

Hierarchical Logit

frequency of the service (recomputed as per hour in a 18 hour travel day and set =1 for car) and Time, the actual travel time on the mode. The characteristics of the traveller are Hinc, household income, and Npers, number of persons in the travelling party. Both characteristics have been assigned to the air mode in this example.

The parameter estimates and the log likelihood values obtained from this model are presented in figure 2. The Asc_air, Asc_tra, Asc_bus are the (NALTS - 1), where NALTS is the number of alternatives, alternative specific constants which can be assigned to the model

Figure 2: HLOGIT Output of MNL Parameter Estimates

INTER-CITY TRAVEL MODE CHOICE				
Parameter Estimates and Log-Likelihoods - MNL model				
Name	Status	Parameter	stand.error	t-statistic
Cost	free	-.006017	.007049	-.853665
termtime	free	-.021389	.010875	-1.966747
hinc_air	free	.002320	.001185	1.966871
npers_air	free	-.495204	.236178	-2.096741
frequent	free	-.228895	.082620	-2.770455
time	free	-.191950	.076549	-2.507529
Asc_air	free	-.138229	1.469666	-.094055
Asc_tra	free	2.269314	.787921	2.880126
Asc_bus	free	.145447	.582218	.249815
Loglikelihood			-125.5502	
Loglikelihood at zero			-163.5827	
Total Number of iterations			5	
Total Number of evaluations			42	
Number of Observations			118	
Mean Square Gradient (-1)			.2777D+02	

The estimation is repeated as a 3 level FIML model with the alternatives grouped as fast (air), and slow (bus, train, car), then further subdivided in the slow mode into public (bus, train) and private (car). Figure 3 shows the HLOGIT instructions to input this model, while figure 4 displays the tree diagram output by the program

Figure 3: HLOGIT Input for a 3 level NMNL Model

```

NMNL
Tree (air) ( ( tra, bus), car)

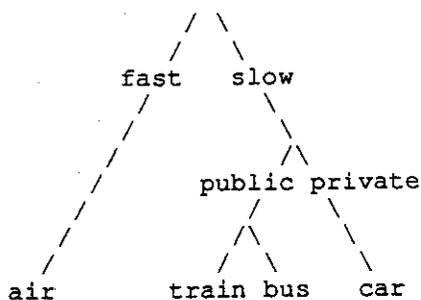
branch = air = fast
branch = (tra, bus) = public
branch = car = private
limb = fas = fast
limb = (pub, pri) = slow

indep = cost, termtime, hinc_air, npers_air, frequent, time

OUTPUT Tree, Predic

```

Figure 4: Tree Diagram for a 3 level NMNL Model



When the model was run, starting with the MNL parameter estimates, we obtained better model results, shown in figure 5. The log likelihood value has improved markedly. The alternative specific constants are no longer statistically significant suggesting there is a far smaller unexplained component in the model. θ_{public} and τ_{slow} are both much closer to 0 than to 1 suggesting the tree structure is justified (dissimilarity parameter values of close to 1 imply the model collapses to MNL). In particular θ_{public} is very significant

Hierarchical Logit

with a t-statistic of 6.4. Since train and bus are more similar to each other than either is to car this result seems reasonable. The independent explanatory variables are also far less significant as more of the choice is explained by the model structure.

Figure 5: NMNL Parameter Estimates from HLOGIT

INTER-CITY TRAVEL MODE CHOICE

Parameter Estimates and Log-Likelihoods - NMNL model

Name	Status	Parameter	stand.error	t-statistic
Cost	free	-.013705	.014028	-.977012
termtime	free	-.011379	.017275	-.658734
hinc_air	free	.004142	.001397	2.963479
npers_air	free	-.881091	.297474	-2.961902
frequent	free	-.109775	.156243	-.702588
time	free	-.116918	.144833	-.807263
Asc_air	free	.056551	1.908749	.029627
Asc_tra	free	1.907287	2.536405	.751964
Asc_bus	free	.972361	1.375046	.707148
theta_pub	free	.097236	.142136	6.391346
tau_slow	free	.296311	.378527	1.859013
Loglikelihood			-96.9558	
Loglikelihood at zero			-163.5827	
Total Number of iterations			5	
Total Number of evaluations			43	
Number of Observations			118	
Mean Square Gradient (-1)			.3884D+01	

The prediction success output in figure 6 shows 65.25% of choices were correctly predicted by the NMNL model. This compared with 53.93% for the MNL model.

Figure 6: Prediction Success Output for 3 level NMNL Model

PREDICTION SUCCESS TABLE

Observed	Predicted Alternative (Max. Prob)			
	Air	Train	Bus	Car
Air	17	2	3	13
Train	2	23	0	3
Bus	3	0	13	4
Car	6	3	2	24

Percent Correctly Predicted 65.25%
 Likelihood at Zero -163.5827
 Likelihood at Actual Choices -96.9558
 Likelihood at Predicted Choices -55.6794
 Utility at Predicted Choices 125.9009

MARKET-SHARES OF ALTERNATIVES

Alt.	Observed Shares and Frequencies		Predicted Shares and Frequencies (Max. Prob) (Mean Choice-Prob.)			
	Air	.2966	35	.2373	28	.2862
Train	.2373	28	.2373	28	.2291	26.19
Bus	.1695	20	.1525	18	.1743	20.57
Car	.2966	35	.3729	44	.3175	37.47

R-SQUARE-EQUIVALENTS

Weights	None	Freq	Probs
Simple	.39978	.42906	1.0000
Efron's	.39086	.41886	.9999
Mcfadden's	.40430		
Successes	.64388	.84984	

CONCLUSION

We believe that this new Hierarchical Logit package will be a valuable addition to the range of software currently available for modelling discrete demand problems. It provides fast and accurate estimation of single level and nested full-information multinomial logit models. Thus it brings sophisticated econometric techniques to the general user rather than keeping them in the realms of research. HLOGIT makes sequential nested logit unnecessary, and MNL programs, such as BLOGIT, obsolete since it runs single level logit so easily. It is, to our knowledge, the first FIML package available with a simple data and model set up interface (Although LIMDEP is a friendly package the user has to write a 'macro' to carry out simultaneous, instead of sequential nested logit estimations)

The current version of HLOGIT is written in Fortran for IBM compatible P.C.s running MS-DOS . A mainframe version is installed on the Macquarie University FACOM computer (similar to IBM mainframes). In keeping with our intention that the package will be accessible to all interested users, we plan future adaptation to other systems. A VAX version will soon be available and a MACINTOSH version is planned for 1990.

This paper has concentrated on expounding the underlying theory which explains the advantage of using NMNL models in many situations, and then emphasised the ease with which models can be set up. For those not familiar with logit programs this emphasis may seem strange, but its importance cannot be overstated. A great deal of specification is needed in this type of model. We have drawn on considerable experience with various logit packages (e.g. XLOGIT, BLOGIT, ALOGIT, LIMDEP, FIML-NL,QUAIL) to devise a simple "fool proof" specification system.

In all transport modelling work "unknowns" abound. Successful solutions depend as much on the expert knowledge of the investigator as on good mathematical models. If one has to pass an analysis on to a colleague who understands the complexities of a logit package but is not familiar with the problem, valuable insights can be lost.

HLOGIT allows the general user to answer most questions of interest about a data set and run models of increasing complexity from least squares regression to Full Information Maximum Likelihood estimation of Hierarchical choice models in a single simple package. We hope it will be used to shed light on many transport demand problems.

Acknowledgments : We are indebted to Axel Boerch-Supan for his collaboration in the creation of this software package, without him HLOGIT would not exist. We thank David Hensher for his advice and suggestions in the preparation of this paper. And we gratefully acknowledge the financial contribution made to partially support this work from the West Australian Department of Transport via John Taplin.

REFERENCES

- Amemiya, T. (1981) Qualitative Response Models, *Journal of Economic Literature*, Vol XIX, pp 1483-1536
- Boersch - Supan, A. (1984) The Demand for Housing in the United States and West Germany : A Discrete Choice Analysis, Unpublished Ph.D. Dissertation Massachusetts Institute of Technology.
- Boersch - Supan, A. (1985) Hierarchical choice models and efficient sampling : with applications on the demand for housing, *Methods of Operations Research*, Vol 50, pp 175-186
- Cosslett, S. (1978) Efficient Estimation of Discrete Choice Models from Choice-Based Samples, Unpublished Ph.D Dissertation, Department of Economics, University of California, Berkley
- Engles, R.F. (1983) Wald, Likelihood Ratio and Lagrange Multiplier Tests in Econometrics, in Griliches Z. (ed.), *Handbook of Econometrics*, Elsevier Press, Amsterdam.
- Goldfeld, S.M. and Quandt, P.E. (1972) *Non Linear Methods in Econometrics*, North Holland Publishing Company, Amsterdam
- Hensher, D.A. and Johnson, L.W. (1981) *Applied Discrete Choice Modelling*, Croom Helm, London (Wiley, New York)
- Hensher, D.A. (1986) Sequential and Full Information Maximum Likelihood Estimation of a Nested Logit Model, *Review of Economics and Statistics*, LXVII (4), 657-667
- Manski, C.F. and Lerman, S.R. (1977) The Estimation of Choice Probabilities from Choice Based Samples, *Econometrica* Vol 45 (8)
- McFadden, D. (1978) Modeling the Choice of Residential Location, in Karlqvist, A., *Spatial Interaction Theory and Residential Location*, North Holland Publishing Company, Amsterdam.
- McFadden, D. (1977) Quantitative Methods for Analysing Travel Behaviour of Individuals : Some Recent Developments ,in Hensher, D.A. and Stopher, P., *Behavioural Travel Modeling* Croom Helm (London)
- McFadden, D. (1983) Qualitative Choice Models, in Griliches Z. (ed.), *Handbook of Econometrics*, Elsevier Press, Amsterdam.