Exercise 4

Stated Choice Experiment

==

This study will replicate the computations in Kenneth Train's stated preference study of California utility customers. The study is discussed in Slides 8-13 in Part 6 of our class notes. The data set,

TrainCalUtilitySurvey.lpj

contains 17,232 rows, 4308 choice tasks in total. There are 4 choices in each task. Individuals made multiple choices, most 12, a few made 8, 9, 10, or 11. There are altogether 361 individuals in the sample. The variables in the data set are

choice	= the dependent variable, indicator for the choice
ntask	= the number of choice tasks, 8-12, repeated for each row of data for the person
price	= price in contract
cntlngth	= contract length
local	= local utility or not (dummy variable)
known	= well known utility (dummy variable)
tod	= time of day rates (dummy variable)
seas	= seasonal rates (dummy variable)

==

The template model command for the multinomial logit model is

NLOGIT ; Lhs = choice ; Choices = Firm1, Firm2, Firm3, Firm4
; Rhs = price, cntlngth, local, known, tod, seas $

Note that although it would be possible to compute alternative specific constants (ASCs), since the choices are not labeled – they are just firm1 to firm4 – it would not make sense to identify the constants with specific firms. Firm1 is 'Firm1' only because it is the first firm in the choice set.

I. Multinomial Logit

Fit the basic MNL model to see what the coefficients look like. To see if customers react to proposed prices, include **;Effects:Price(*)** in the equation. This requests a report of the elasticities.

II. Mixed Logit

In Train's study, a random parameters (RP) model was used with the following model assumptions:

Normal:	cntlngth, local, known
Lognormal:	tod, seas
Fixed:	price.

In addition, it is known (strongly believed) that the lognormal coefficients are negative. The lognormal distribution is used to keep the coefficients on one side of zero. To force them to be negative, rather than positive, we put the minus sign on the variable by using the negative in the model. Use

CREATE ; mtod = -tod ; mseas = -seas $

To fit an RP model, we add to the basic command

; RPL ; Pds = ntask ; Halton ; Pts = 50 ; Parameters
; Fcn = cntlngth(n), local(n), known(n), tod(l), seas(l)

Note that *tod* and *seas* must be changed to *mtod* and *mseas* in the **;Rhs specification**. The **;Parameters** is going to capture the 'individual specific' estimates of the parameters. The **;Fcn specification** specifies which parameters are random and what distributions are to be used. The '(n)' specifies a normally distributed parameter, '(l)' requests lognormal.

a. Fit the RP model. How closely do your results resemble the published results?(They are shown in slide 10 of part 6.) (The results are not replicated exactly because Train used a different program (Gauss), a different random number generator (note we are not actually using random draws – Halton sequences are not random) and probably a much large number of draws for the simulation.) Do the estimated elasticities resemble those for the multinomial logit model?

b. The estimated parameters for the two lognormal results are the underlying normal mean and standard deviation. The population mean for the lognormal coefficient is $\exp(\mu + \sigma^2/2)$ where μ and σ are the two estimated parameters for the model. Does the result resemble the result for the original MNL model? Note, NLOGIT has a calculator you can use for this. Use

CALC ; List ; exp (b(4) + .5*b(10)^2) $
(This is for the coefficient on TOD. Change b(4) and b(10) to b(5) and b(11), respectively for SEAS.)

Before fiting the triangular model next, use the command

MATRIX ; bl = beta_i $

(This will save the matrix before the next model command overwrites it.)

c. The lognormal model is often difficult to estimate because the distribution has a long thick tail and allows extreme values of the parameters. An alternative often used is the triangular distribution, which has a finite range and still maintains the one sidedness of the lognormal model. To fit the triangular model, change '(l)' to '(o)' in the model command – this form requests a triangular with one leg anchored at zero. Use

MATRIX ; bt = beta_i $

to capture the random parameters matrix.

d. The next operation will explore the person specific variation in the parameters. We will compute E[b(i,price)|All data for individual i] for each individual in the sample under the two model specifications (lognormal and triangular). We will then plot kernel estimators to compare these two distributions. The commands for this study are given in the command file for this exercise.

III. Latent class models

The latent class model is an alternative way to specify random parameter variation. For this application, the template model command would be

```
NLOGIT    ; Lhs = choice ; Choices = Firm1, Firm2, Firm3, Firm4
          ; Rhs = price, cntlngth, local, known, tod, seas
          ; LCM ; Pds = ntask  ; Pts = number of classes $
```

We have no priors on the number of classes, or even if this is an appropriate model for these data. Ignoring that for now, we consider a search for the number of classes. A common way to approach this question is to use the AIC, which is reported with the model results. The lower the AIC, the better.

a. Fit an LCM with 2 latent classes. Then with 3 classes, then 4. Which seems to be the best model?

b. I have a theory that there are two types of customers. Some care about time of day and seasonal rates and some do not. In the context of the LCM, this would be two classes, one in which the coefficients on tod and seas are nonzero and another in which they are zero. Latent class models such as these are obtained by imposing constraints on the class specific parameters. For this theory, there are two possibilities shown in the following template: In the first specification, there are two different classes of individuals. The taste parameters are different in the two classes, and in one, tod and seas do not enter the utility functions. In the second specification, the model parameters are the same in the two classes, but in one of the two classes, the coefficients on tod and seas are zero. This second specification is a 'nonattendance' model. Note how ;RST (restrictions) is used to define the model structure.

```
? Two classes, one with nonattendance to tod and seas
NLOGIT    ; Lhs = choice ; Choices = Firm1, Firm2, Firm3, Firm4
          ; Rhs = price, cntlngth, local, known, tod, seas
          ; LCM ; Pds = ntask ; Pts = 2
          ; RST = b1,b2,b3,b4,b5,b6, c1,c2,c3,c4,0,0 $
? Attribute nonattendance
NLOGIT    ; Lhs = choice ; Choices = Firm1, Firm2, Firm3, Firm4
          ; Rhs = price, cntlngth, local, known, tod, seas
          ; LCM ; Pds = ntask ; Pts = 2
          ; RST = b1,b2,b3,b4,b5,b6, b1,b2,b3,b4,0,0 $
```

c. In the second model above, choosers are otherwise the same save for their preferences over the rate types. A fuller specification of this way of thinking is the 2^K model, which considers all the possible configurations of attribute nonattendance,

```
NLOGIT    ; Lhs = choice ; Choices = Firm1, Firm2, Firm3, Firm4
          ; Rhs = price, cntlngth, local, known, tod, seas
          ; LCM ; Pds = ntask ; Pts = 4
  ; RST = b1,b2,b3,b4,b5,b6,   b1,b2,b3,b4,0,0,   b1,b2,b3,b4,b5, 0,  b1,b2,b3,b4,0, b6 $
```

Using these specifications, develop a nonattendance model. Use the likelihood ratio test to see if the attribute nonattendance model in any of these forms is a reasonable specification for our data