Computer Inputs for the Nevo BLP Example (31 December 2005. 19 January 2006)

Listed below is a Matlab script file and eight accompanying Matlab functions and a number of data files that compute the random coefficients discrete choice model described in Aviv Nevo's article,

Nevo, Aviv (2000) "A Practitioner's Guide to Estimation of Random Coefficients Logit Models of Demand," *Journal of Economics & Management Strategy* 9(4): 513-548, 2000.

You can download everything as a zip file by downloading nevofiles.zip.

The code has been provided for teaching by Aviv Nevo (and modified by Bronwyn Hall and Eric Rasmusen to run in Matlab 7). Users of this code should reference the Nevo paper, since the code was constructed for the appendix to it. Eric Rasmusen has run this code with partial success using Matlab 6.5.0 on the libra.uits.iu.edu computer with its Unix operating system. For questions regarding Matlab, go to MathWorks.

I could not get the quasi-Newton algorithm to work properly with the Jacobian, so in the code r1.m below I turned off the gradient option. Also, I have not run the simplex algorithm to completion, only to 150 iterations in the code below. Thus, it may be better to depend on the earlier code by Nevo and Hall.

The program consists of the following Matlab m-files:

- r1.m - A script file that reads in the data and calls the other functions. This is the main program. Matlab works by putting functions in separate programs, which is why there are so many files. This is the file called rc_dc.m by Nevo and Hall.
- gmmobjg.m - This function computes the GMM objective function and its gradient. It replaces two of Nevo's files, gmmobj.m and gradobj.m.
- meanval.m - This function computes the mean utility level.
- mufunc.m - This function computes the non-linear part of the utility (mu_ijt in Nevo's "Guide").
- mktsh.m - This function computes the market share for each product.
- ind_sh.m - This function computes the "individual" probabilities of choosing each brand.
- jacob.m - This function computes the Jacobian of the implicit function that defines the mean utility. I could not get the program to make effective use of this file, but I include it anyway.
- var_cov.m - This function computes the VCov matrix of the estimates.
- cd_dum.m - This function creates a set of dummy variables.

The results Nevo obtained with the code can be found in results-nevo.txt. The results Rasmusen obtained can be found in results-newtonraphson.txt and results-simplex.txt for the two different matlab optimization routines that can be used.

The files just named contain only the results. For logs of how the program ran, see mydiary-newtonraphson.txt and mydiary-simplex.txt.

The code needs data files as input. These data were motivated by real scanner data, but it is not real and should not be used to make any inference. The purpose these data serve is to provide an example of the inputs required by the program. The data consists of two Matlab files: ps2.mat and iv.mat (both Matlab 5+ files). If you use Matlab, use these files directly. If you use some other optimization program, either use the

Matlab "load" and "save" commands to create ASCII files or download the Excel spreadsheets that contain the data (cereal_ps3.xls and demog_ps3.xls).

The date are (semi-fabricated) data on 24 brands of the only REAL product (ready-to-eat cereal, what else did you think?), for 94 markets (47 US cities for the first 2 quarters of 1988). These variables are defined and were treated as described in Nevo (2000).

The file ***ps2.mat*** contains the following variables:

id - an id variable in the format bbbbccyyq, where bbbb is a unique 4 digit identifier for each brand (the first digit is company and last 3 are brand, i.e., 1006 is K Raisin Bran and 3006 is Post Raisin Bran), cc is a city code, yy is year (=88 for all observations is this data set) and q is quarter. All the other variables are sorted by date city brand.

id_demo - an id variable for the random draws and the demographic variables, of the format ccyyq. Since these variables do not vary by brand they are not repeated. The first observation here corresponds to the first market, the second to the next 24 and so forth.

s_jt - the market shares of brand *j* in market *t*. Each row corresponds to the equivalent row in ***id***.

x1 - the variables that enter the linear part of the estimation. Here this consists of a price variable (first column) and 24 brand dummy variables. Each row corresponds to the equivalent row in ***id***. This matrix is saved as a sparse matrix.

x2 - the variables that enter the non-linear part of the estimation. Here this consists of a constant, price, sugar content and a mushy dummy, respectively . Each row corresponds to the equivalent row in ***id***.

v - random draws given for the estimation. For each market 80 iid normal draws are provided. They correspond to 20 "individuals", where for each individual there is a different draw for each column of ***x2***. The ordering is given by ***id_demo***.

demogr - draws of demographic variables from the CPS for 20 individuals in each market. The first 20 columns give the income, the next 20 columns the income squared, columns 41 through 60 are age and 61 through 80 are a child dummy variable (=1 if age <= 16). Each of the variables has been demeaned (i.e. the mean of each set of 20 columns over the 94 rows is 0). The ordering is given by ***id_demo***.

The file ***iv.mat*** contains the variable ***iv*** which consists of an id column (see the ***id*** variable above) and 20 columns of IV's for the price variable. The variable is sorted in the same order as the variables in ***ps2.mat***.

Excel spreadsheets:

cereal contains 2256 observations on id, brand, firm, city, quarter, share, price, sugar content, mushiness, and the 20 instruments in ***iv***, called ***z1-z20***.
demog contains the demographic draws for each market. There are 94 observations (47 cities by 2 quarters) and 80 variables (20 individuals X 4 variables).