SUPPLEMENT TO "BEST NONPARAMETRIC BOUNDS ON DEMAND RESPONSES"—GUIDE TO DATA FILES AND PROGRAM

(*Econometrica*, Vol. 76, No. 6, November 2008, 1227–1262)

BY RICHARD BLUNDELL, MARTIN BROWNING, AND IAN CRAWFORD

1. OVERVIEW

THE MAIN DATA FILES are in GAUSS format. The program files are a mixture of Gauss and MatLab scripts which include a few procedure files. The GAUSS files generally handle estimation; the MatLab files generally handle numerical problems. Intermediate data sets are generated in the course of running the programs and all of the file paths currently in the programs refer to our own LAN. All of the program files have been annotated at key points and the descriptions below refer to the points in the paper where they are used.

The general sequence for running the files is as follows:

1. The GAUSS program **Englecurves.prg** estimates the semiparametric Engel curves as described in Section 3.2.1 in the paper. It produces some intermediate GAUSS fmt format data files which contain the various components of the regressions plus some descriptive statistics which are used when defining the price–budget combinations at which the demand curves will be evaluated.

2. The GAUSS program **Interceptdemands.prg** picks up the regression components produced by Engelcurves.prg and evaluates the intercept demands (defined in Section 2.1, Definition 4 in the paper). The user defines the price–budget combinations at which they want to bound the demand curves in this program. It produces ASCII text files containing the intercept demands on each Engel curve, their variance–covariance matrix, the prices, the new price–budget combinations, and so forth.

3. The MatLab program **DemandCurvePerturbandDrop.m** picks up the ASCII files generated by Interceptdemands.prg and generates the demand curve bounds. It also creates the data and draws the incidental Figures 3 and 7. The program allows the user to either drop periods which violate SARP (this will create Figures 4, 5, and 6) or to impose SARP on all the intercept demands (Figures 8, 9, and 10) or to impose SARP on a subset of periods (Figures 12 and 13).

2. LIST OF FILES AND BRIEF DESCRIPTIONS

2.1. *Data Sets*

data75.dat/dht... data99.dat/dht. These are 25 GAUSS data files arranged by FES year.

pall.fmt is a GAUSS fmt format data file containing disaggregated price data by year.

 DOI: 10.3982/ECTA6096SUPP

2.2. *Procedures*

2.2.1. *GAUSS*

garpproc.prc is a library of RP-related GAUSS procedures (not all are called).

2.2.2. *MatLab*

garp.m tests GARP (used in emax.m).

emax.m computes the maximum Afriat efficiency at which a data set satisfies GARP (used in GARPConstraints.m).

WeightedEuclideanDistanceKminus1.m computes the distance metric which serves as the objective function in the imposition of the RP condition on the intercept demands (Equation (12), Section 5.1).

GARPConstraints.m provides the nonlinear constraints for the same problem (Equation (12), Section 5.1).

2.3. *Scripts/Programs*

2.3.1. *GAUSS*

prices.prg generates and saves the annual price data.

englecurves.prg estimates the semiparametric Engel curves as described in Section 3.2.1 in the paper. It produces some intermediate GAUSS fmt format data files which contain the various components of the regressions plus some descriptive statistics which are used when defining the price–budget combinations at which the demand curves will be evaluated.

interceptdemands.prg picks up the regression components produced by Engelcurves.prg and evaluates the intercept demands (defined in Section 2.1, Definition 4 in the paper). The user defines the price–budget combinations at which they want to bound the demand curves in this program. It produces ASCII text files containing the intercept demands on each Engel curve, their variance–covariance matrix, the prices, the new price–budget combinations, and so forth.

resample.prg is a version of engelcurves.prg which is set in a resample loop. It generates a user-defined number of resampled Engel curve estimates at a user-defined sample rate and saves the results in ASCII format.

2.3.2. *MatLab*

DemandCurvePerturbandDrop.m picks up the ASCII files generated by *Interceptdemands.prg* and generates the demand curve bounds. It also creates the data and draws the incidental Figures 3 and 7. The program allows the user to either drop periods which violate SARP (this will create Figures 4, 5, and 6) or to impose SARP on all the intercept demands (Figures 8, 9, and 10) or to impose SARP on a subset of periods (Figures 12 and 13).

Resampling.m picks up the ASCII files generated by *resample.prg* and generates the resulting empirical distribution of the loss function described in Equation (12), Section 5.1.

Dept. of Economics, University College London, Gower Street, London WC1 E6BT, U.K. and Institute for Fiscal Studies, London, U.K.; r.blundell@ucl.ac.uk,

Dept. of Economics, Oxford University, Oxford, OX1 3UQ and Institute for Fiscal Studies, London, U.K.; Martin.Browning@economics.ox.ac.uk,

and

Dept. of Economics, Oxford University, Oxford, OX1 3UQ and Institute for Fiscal Studies, London, U.K.; ian.crawford@economics.ox.ac.uk.

Manuscript received October, 2005; final revision received December, 2007.