Advanced Microeconomics

Introduction to GAMS

Jan Hagemejer

October 23, 2012

The software

- We will use GAMS - General Algebraic Modeling System, a high-level modeling system for mathematical programming problems.
- It is commercial software, however it can run small models in the free version.
- It can solve linear and non-linear mathematical programming problems.
- It is very useful as it allows the user to use the indexed notation, for example $X(L, I)$ may be a variable x that is indexed over households I and goods L.

The GAMS IDE

- Normally you will use the GAMS IDE or the development interface for GAMS.
- Once you fire it up for the first time, it will ask you about default solvers. Just accept the default.
- You have to start a new project (File/Project/New Project) in a directory of your choice (eg. in your desktop etc)
- GAMS is installed in a computer lab - if you use it there - place your project on your thumbdrive or in your network directory.
- Once you are in a project, you can either start a new program file or open an example program from my website which you should place in the project directory.
- GAMS program files have .gms extension. GAMS output (results) files have .lst extension. Both are standard text files.
- All GAMS statements should end with a semi-colon ";".

Some syntax

- GAMS originates from old-time mainframes with text terminals with limited character sets, therefore some strange syntax remains:
  - ** is a power
  - =G=, =E=, =L= are inequality signs
- $SUM(L, X(L))$ - this command sums up all the X's from a set L - $\sum_{l=1}^{L} x_l$.
- $PROD(L, X(L))$ - this command takes a product of all the X's from a set L - $\prod_{l=1}^{L} x_l$.

Sets, parameters

SETS Declares the indexes of the model. If your model uses sets, your file has to start with a declaration of the sets

```
SETS
    L /X1, X2/;
```

PARAMETERS Declares the constants or parameters to be used in the program. You can insert a verbal description and you can put a value when declaring a parameter or put it later in the code using A=5 syntax.

```
PARAMETERS
ALPHA(L) Parameter of the CD function /X1 0.3, X2 0.7/
A(L) Parameters of the Leontief function /X1 3, X2 1/;
```

If you want to refer to a particular element of the set, use inverted commas: eg. A(X1)

Variables

VARIABLES Have to be declared as well. Variables can be exogeneous or endogeneous

```
VARIABLES
X(L) Demand for good L
U    Utility level
P(L) Prices
W    Income;
```

- If a variable is to be exogeneous (fixed), you fix it with .FX suffix. Example: W.FX=100;
- If a variable is endogeneous, you need to have the same number of equations as you have endogeneous variables.

Equations

Once variables are declared, we have to declare equations that we are goin to use:

```
EQUATIONS
BC        Budget constraint
IC(L)     Inequality constraints
CD        Cobb-Douglas utility function LEO
Leontieff utility function
UPS       Perfect subs. linear utility function ;
```

Then, we need to type the equation, using the equation name first. Note the $= E =$ equality and $= G =$ and $= L =$ inequality sign

```
BC.. SUM(L, P(L) * X(L)) =L= W;
IC(L).. X(L) =G= 0;
CD.. PROD(L, X(L)**ALPHA(L)) =E= U;
LEO.. MIN(X("X1")/A("X1"), X("X2")/A("X2") ) =E=U;
UPS.. SUM(L, B(L) * X(L)) =E= U;
```

We fix variables that are exogeneous:

```
P.FX("X1") = 1; P.FX("X2") = 2; W.FX = 100
```

Model declaration and solution. We need to declare what equations will be included in our model:

```
MODEL CONSUMER
/
BC
CD
IC
/;
```

Then we solve the model by showing gams what variable to maximize / minimize:

```
Solve CONSUMER USING DNLP MAXIMIZING U;
```

And display the levels of variables in the model.

```
DISPLAY W.L, P.L, X.L, U.L;
```

Results

- If we did not make mistakes, GAMS will solve the model and open the .lst file with results in another window.
- On the left handside you will find a browser guides you through the solution file
- "Solution report" gives you the status of solution
- "SolVAR" shows levels of variables
- "Display" shows everything you asked GAMS to display
- If you solved the model more than once, this parts will reappear in the .lst file for each time you solved the model.

Things to do

- Solve the Cobb-Douglas consumer problem with income equal to 100 and prices 1 and 2.
- Compute the price elasticity of demand by increasing the prices and comparing the solutions.
- Compute the income elasticity of demand in the same way
- Repeat the above for the Leontieff consumer
- Play with prices of goods in the perfect substitutes case
- Rewrite the problem of perfect substitutes as a Kuhn-Tucker problem using MCP solver (second file).

With Mixed Complementarity Problem we write the problem as a system of Kuhn-Tucker conditions. To each equation we attach a shadow-price (eg. a Lagrange multiplier)

```
FOCO..       W - SUM(L, P(L) * X(L)) =g= 0;                    ;
FOCL(L)..    X(L) =g= 0;
FOCX(L)..    B(L) -  P(L)*LAMBDA0 + LAMBDA(L) =E= 0 ;
```

The model statement now includes the equation-shadow-price pair:

```
MODEL CONSUMER
/
FOCO.LAMBDA0
FOCL.LAMBDA
FOCX.X
/;
```