```
$oninline
/*

GAMS program used to estimate capacity with the hyperbolic graph efficiency measure, with variable returns to scale
and undesirable outputs.

Source: Färe, R., J.E.  Kirkley, and J.B. Walden. 2007. "Estimating Capacity and Efficiency in Fisheries with
Undesirable Outputs." VIMS Marine resource Report N. 2007-6. August 2007.

Author:  John B. Walden
        NMFS/NEFSC
        166 Water St.
        Woods Hole, MA  02543
        (508) 495-2355
        John.Walden@Noaa.Gov


*/

/* The following line turns off listing of some elements in the GAMS listing file*/

$OFFSYMLIST OFFSYMXREF OFFUELLIST OFFUELXREF

OPTION SOLPRINT=ON, SYSOUT=OFF, LIMROW=0, LIMCOL=0;

/* NEXT DEFINE INPUTS AND OUTPUTS.*/


SET INOUT /fix1*fix3, var1*var3, out1*out8/
   OUTPUT(INOUT) /out1*out8/
   Goutput(OUTPUT) /out1*out6/
   Boutput(OUTPUT) /out7, out8/
   FIXED(INOUT) /fix1*fix3/
   VAR(INOUT) /var1*var3/
   OBS /1*1000/

   SUBOBS(OBS) /1*102/

   ACTOBS(OBS)
;

/* OBS sets up memory for 1000 observations.  SUBOBS is the actual number of
observations in the data set*/

/*Next, define an alias for the set SUBOBS */

alias (subobs, subobs1)


$OFFLISTING

TABLE ACT(OBS,INOUT)  INPUT OUTPUT TABLE
$ondelim
$INCLUDE "disc1.csv"
$offdelim
```

```
$ONLISTING

/*Reads in a csv formatted data set.  */


VARIABLES

 lambda     efficiency score
 weight(obs)  weights
 gamma(obs,var)
;

 POSITIVE Variable weight, gamma;



 EQUATIONS

   CONSTR1(GOUTPUT, OBS)  DEA constraint for each output
   CONSTR2(BOUTPUT, OBS)  DEA constraint for BAD Outputs
   CONSTR3(FIXED, OBS)    DEA constraint for Fixed Inputs
   CONSTR4(VAR,OBS)       DEA Constraint for Variable Outputs
   CONSTR5                DEA Constraint for Variable returns to Scale;

   CONSTR1(GOUTPUT, ACTOBS).. SUM(SUBOBS, WEIGHT(SUBOBS)*ACT(SUBOBS,GOUTPUT)) =G=
                              LAMBDA*ACT(ACTOBS,GOUTPUT);

   CONSTR2(BOUTPUT, ACTOBS).. 2*ACT(ACTOBS,BOUTPUT)-LAMBDA*ACT(ACTOBS,BOUTPUT)
                             -SUM(SUBOBS,WEIGHT(SUBOBS)*ACT(SUBOBS,BOUTPUT))  =E= 0;

   CONSTR3(FIXED, ACTOBS).. SUM(SUBOBS,WEIGHT(SUBOBS)*ACT(SUBOBS,FIXED))
                            -ACT(ACTOBS,FIXED) =L= 0;

   CONSTR4(VAR, ACTOBS).. SUM(SUBOBS, WEIGHT(SUBOBS)*ACT(SUBOBS,VAR))
                          -Gamma(actobs,var)*ACT(ACTOBS,VAR) =L= 0;

   CONSTR5.. SUM(subobs, weight(subobs)) =E= 1;


/*Define a parameter to hold results for each pass through the loop*/


 PARAMETER

   score1(obs)  efficiency scores
;

/*Define an external file to hold results which tell whether model solved at each iteration*/

file primal2 /graph_cap_res.txt/

/* The file graph_cap_res.txt holds information for each pass through the loop
so you know the model solved at each iteration*/

 MODEL CAP /ALL/;
```

```
cap.solprint=2;
cap.solvelink=2;

LOOP(SUBOBS1,


     ACTOBS(OBS)=NO;
     ACTOBS(SUBOBS1)=YES;

    SOLVE CAP maximizing lambda USING LP;


     score1(SUBOBS1) = lambda.l

    put primal2;

     if ((cap.modelstat eq 1 and  cap.solvestat eq 1),

       put @1, subobs1.tl, @10, "optimal", @20, "normal completion" /

     else

       put @1, subobs1.tl, @10, cap.modelstat:>2:0,
           @20, cap.solvestat:>2:0/
     );



);



/*The next file is to output results to a file to be imported into a spreadsheet program.  Results could
also be printed to the listing file with the use of the display command*/

file res /'graph_vrs_wd.csv'/ ;

res.pc=5;
res.pw=500;

put res;

put 'Obs','LAMBDA';


put //

 loop (subobs1,

    put /

      put subobs1.tl,score1(subobs1):5:2

       );
putclose;
```