HITS program for Maximum Score Estimation

Kostas Florios

Department of International and European Economic Studies, Athens University of Economics and Business,

cflorios@aueb.gr

Description

Hyperplanes Intersection Tabu Search (HITS) is a program that calculates the Maximum Score Estimator (MS) of the binary choice model (Manski, 1975; 1985). It is a variant of Tabu Search (TS) method (Glover, 1989) especially modified for MS. Specifically, it uses the gaussian elimination to locate the trial points of hyperplanes intersections (Pinkse, 1993). It uses the concept of "Neighbourhood" which is popular in trajectory-based methods of combinatorial optimization in order to search nearby solutions of the current solution and avoid complete enumeration. It also uses a "Tabu List" as a short-term memory of prohibited neighbouring solutions to which the search is prevented from moving to. **The program is coded in Microsoft Visual Studio 2017 with Intel Visual Fortran**.

Input

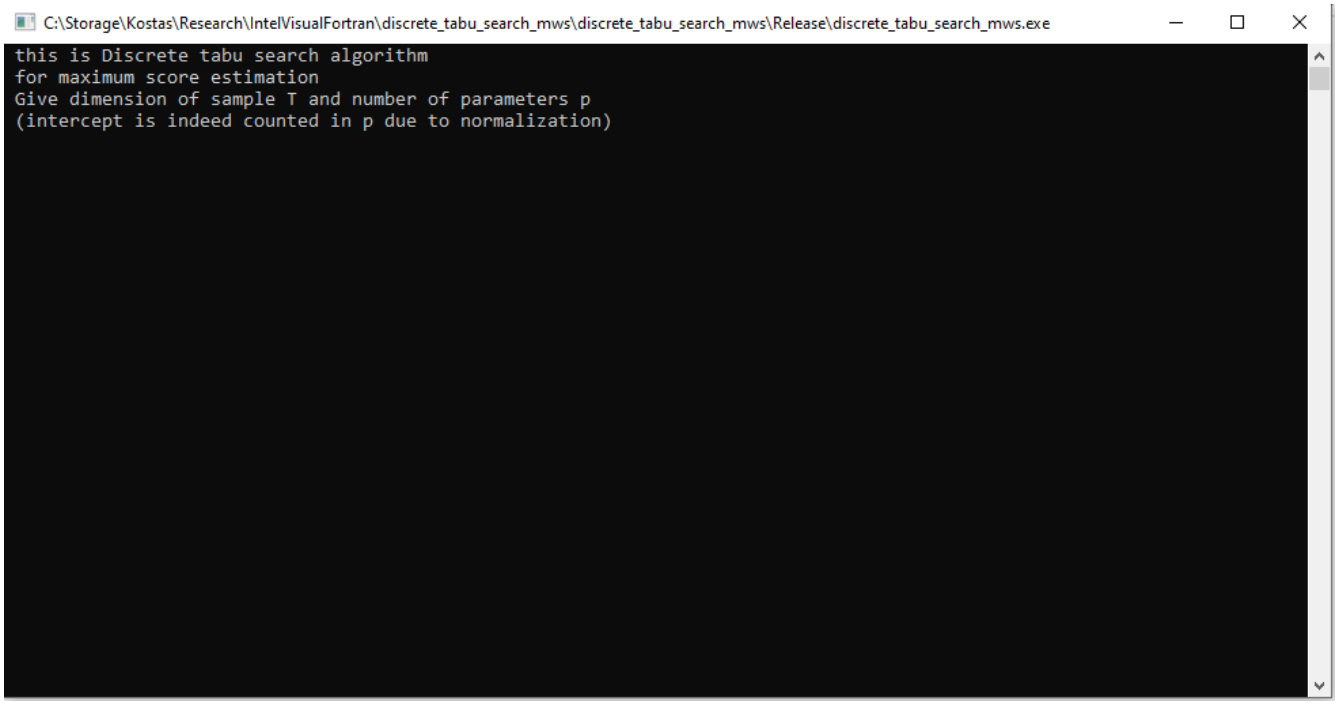
The basic input of the HITS program is 2 text files that contain the required information. Assume that we have the following dataset in a file called Horowitz93fortran_z_intcpt.txt:

1	33	10	36	2	1
2	-69.5	0	3	1	1
3	-41	4	14	0	1
4	-40.5	1	0	0	1
5	21	21	24	0	1
…					
838	-32	10	21	2	1
839	89	30	37	0	1
840	34	11	43	2	1
841	13	-4	40	2	1
842	-100	8	22	2	1

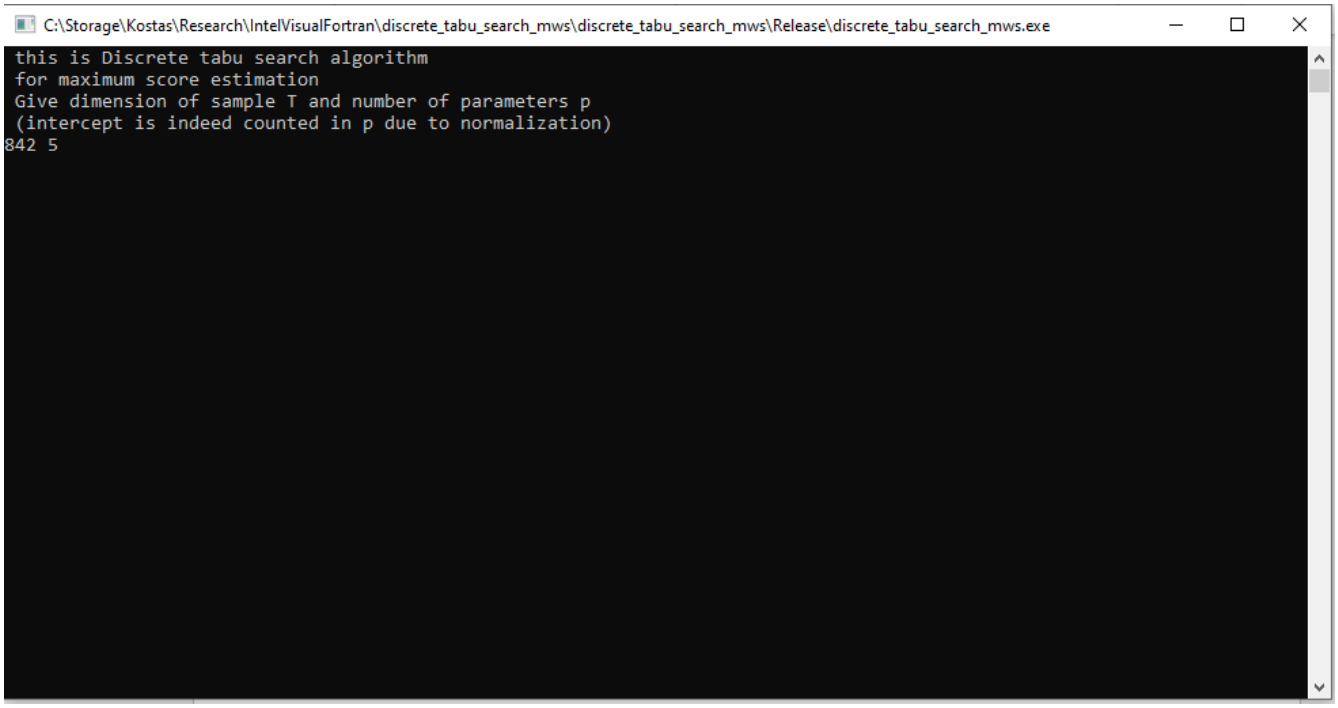
Also, assume the response variable saved in file Horowitz93fortran_y.txt as follows:

1	1
2	-1
3	1
4	-1
5	1
…	
838	1
839	1
840	1
841	1
842	1

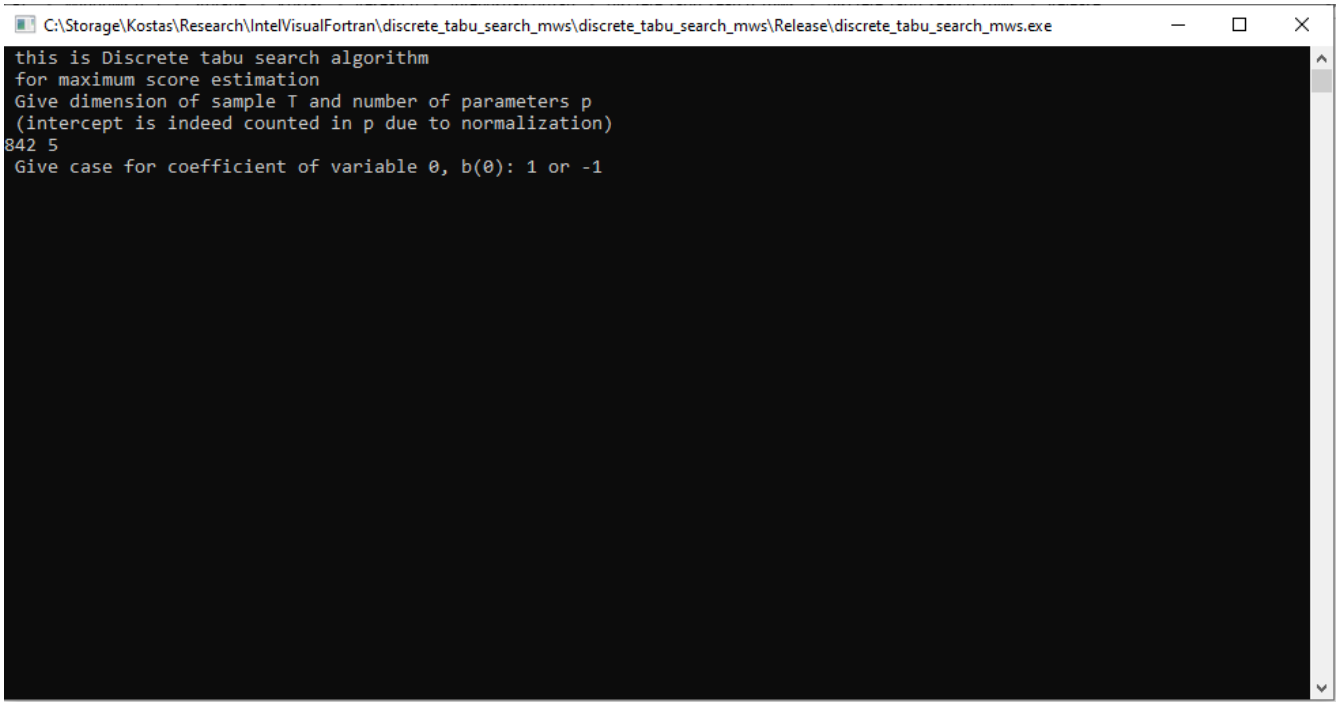
The user is welcome to double click the executable application discrete_tabu_search_mws.exe. The following screen is displayed with the user prompted to input a few parameters first:



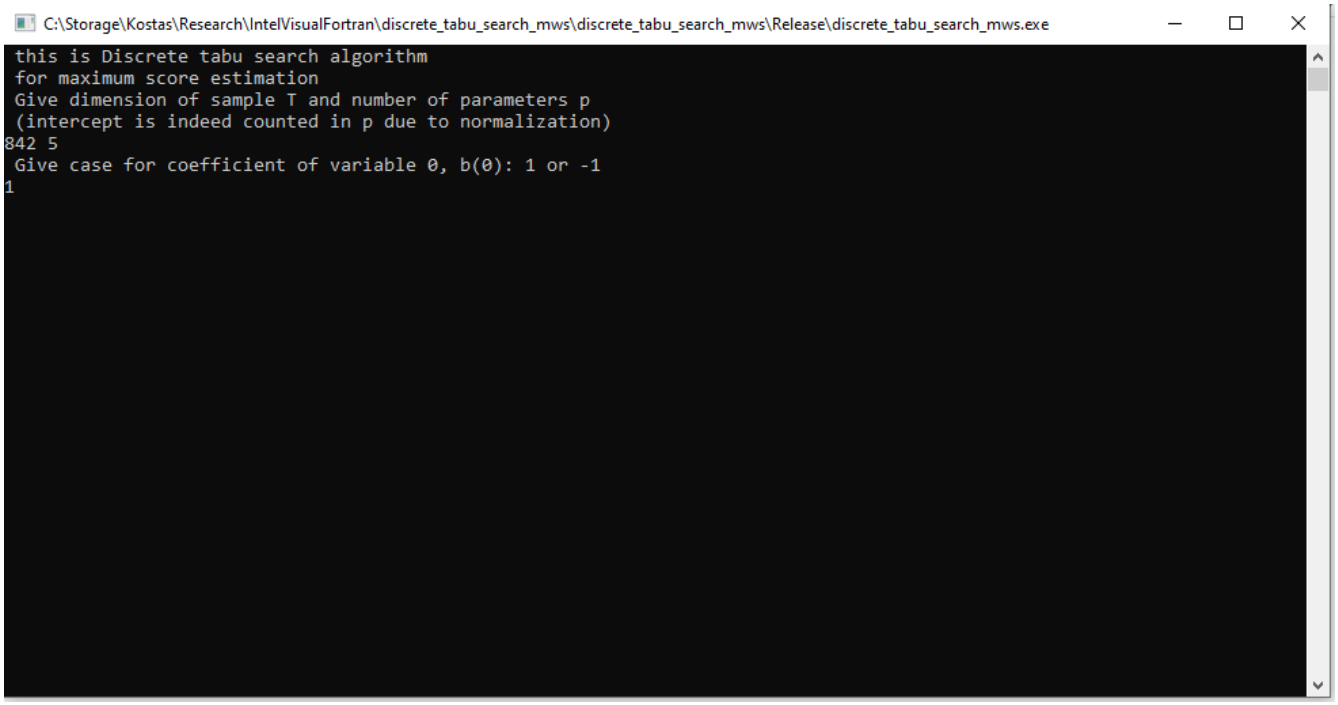
If we count the rows and columns of the dataset in Horowitz93fortran_z_intcpt.txt we can say that the correctly specified values for T and p are 842 and 5, respectively, so we proceed like this:



Hitting enter, the system requires the value for b0 to be set a priori to either 1 or -1:



Here we set the value of b0 to 1 (positive effect of the variable in first column, economic theory specifies this):

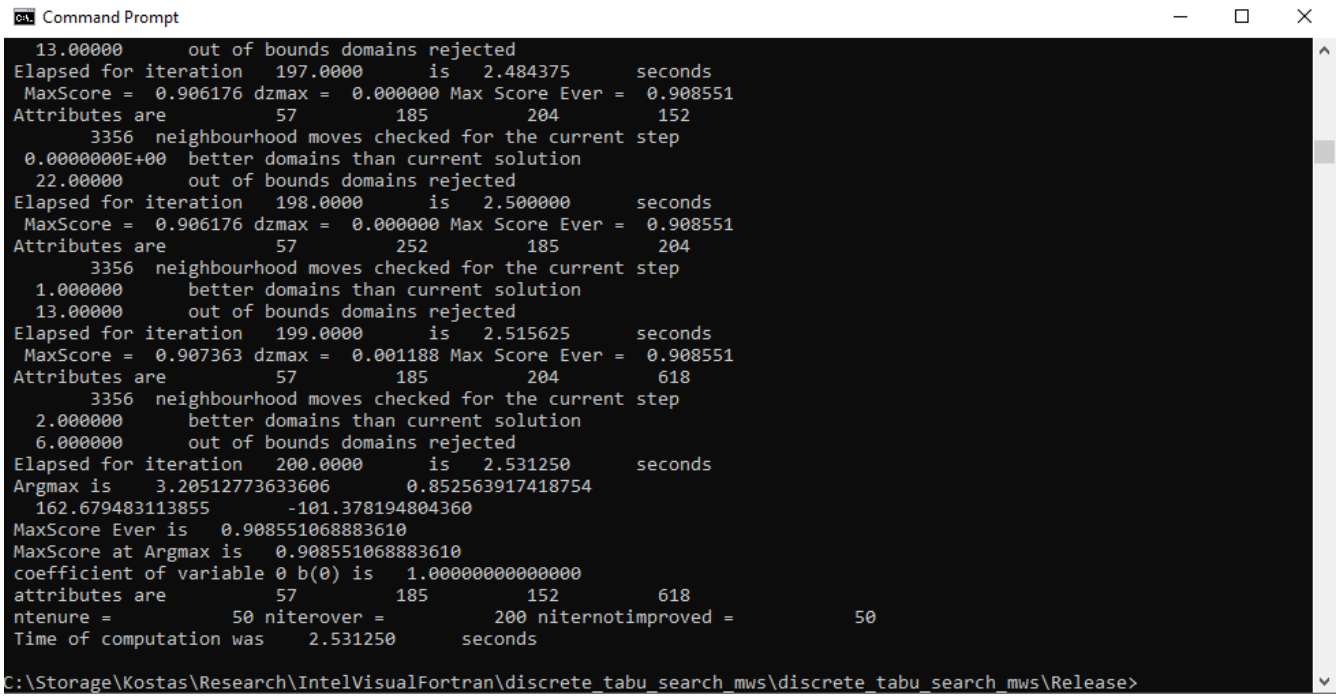


After hitting enter again, the optimization runs.

It is actually very fast since it takes little less than 3 seconds. The final screen looks like this:

Running

During the run the user can see:



Output

The basic output of HITS is a text file with name Horowitz93fortran_tabusearch_restart1_opt_bounds_1e4_log_iter200.txt that contains all the information:

this is Discrete tabu search algorithm
for maximum score estimation
Initial score is 0.388361045130641
Initial attributes 83 100 599 827
MaxScore = 0.745843 dzmax = 0.357482 Max Score Ever = 0.745843
MaxScore = 0.768409 dzmax = 0.022565 Max Score Ever = 0.768409
MaxScore = 0.794537 dzmax = 0.026128 Max Score Ever = 0.794537
MaxScore = 0.808789 dzmax = 0.014252 Max Score Ever = 0.808789
MaxScore = 0.826603 dzmax = 0.017815 Max Score Ever = 0.826603
…
MaxScore = 0.904988 dzmax = -0.001188 Max Score Ever = 0.908551
MaxScore = 0.906176 dzmax = 0.001188 Max Score Ever = 0.908551
MaxScore = 0.906176 dzmax = 0.000000 Max Score Ever = 0.908551
MaxScore = 0.906176 dzmax = 0.000000 Max Score Ever = 0.908551
MaxScore = 0.907363 dzmax = 0.001188 Max Score Ever = 0.908551
Argmax is 3.20512773633606 0.852563917418754
162.679483113855 -101.378194804360
MaxScore Ever is 0.908551068883610
MaxScore at Argmax is 0.908551068883610
coefficient of variable 0 b(0) is 1.00000000000000
attributes are 57 185 152 618
ntenure = 50 niterover = 200 niternotimproved = 50
Time of computation was 2.531250 seconds

The final result is taken by inspecting the lines

```
MaxScore Ever is   0.908551068883610
 MaxScore at Argmax is   0.908551068883610
```

The coefficients are taken from the line

```
Argmax is    3.20512773633606      0.852563917418754
  162.679483113855      -101.378194804360
```

The first coefficient is 1.0 (we set it like that in the beginning in the second question of the software). Then the remaining coefficients are 3.20512773633606 0.852563917418754 162.679483113855 -101.378194804360 as they come in the columns of Horowitz93fortran_z_intcpt.txt. Note that in Horowitz93fortran_z_intcpt.txt the constant term is included in the last column.

The following information is used for debugging:

```
attributes are        57       185       152       618
 ntenure =         50 niterover =         200 niternotimproved =         50
 Time of computation was    2.531250     seconds
```

The computation time was 2.53 seconds and the attributes 57 185 152 618 are the observations id's (numbers from 1 to 842) that define the linear system p×p that is equivalent to the final solution of HITS.

Acknowledgements

I would like to thank Dr. **Alexandros Louka** and Professor **Yannis Bilias** for their valuable help and the shared ideas for the development of HITS. The development of the HITS software was supported by the Hellenic Foundation for Research and Innovation (H.F.R.I.) under the '2nd Call for H.F.R.I. Research Projects to support Post-Doctoral Researchers' (Project Number: 902).

References

C. F. Manski, Maximum score estimation of the stochastic utility model of choice, Journal of Econometrics 3 (1975) 205–228.

C. F. Manski, Semiparametric analysis of discrete response: Asymptotic properties of the maximum score estimator, Journal of econometrics 27 (3) (1985) 313–333.

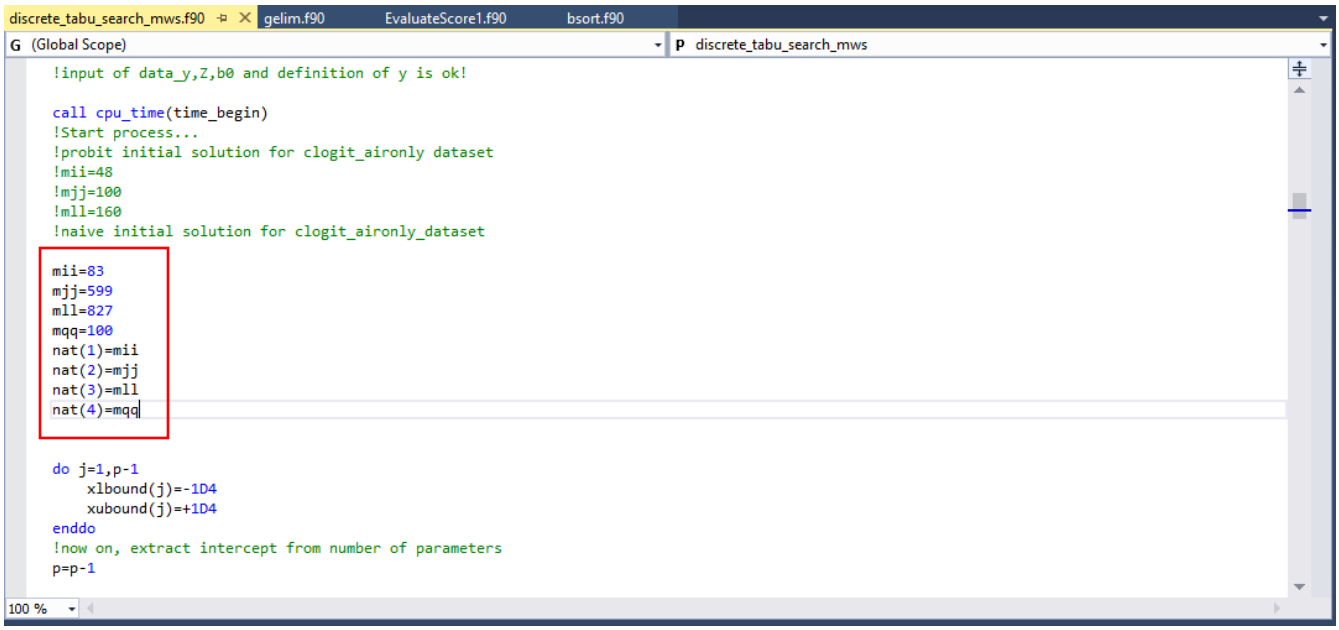
C. A. Pinkse, On the computation of semiparametric estimates in limited dependent variable models, Journal of Econometrics 58 (1993) 185–205.

Glover, F., 1989. Tabu search—part i. ORSA Journal on Computing 1, 190–206.

Advanced Usage

To use for another dataset the user should edit a few lines in the Fortran code and recompile.

Step 1. Locate lines 105-112 in file discrete_tabu_search_mws.f90 and change the hard coded values into what is needed for your own dataset



This is the initialization of the array nat for the starting combination of hyperplanes. In the Horowitz dataset, we need 4 observations as hyperplanes, and we arbitrarily have set

nat(1)=83

nat(2)=599

nat(3)=827

nat(4)=100

The user should define (p-1) nat elements with random integers between 1 and T (different to each other).

Step 2. Locate lines 115-118 of file discrete_tabu_search_mws.f90. The user can change the domain of the optimization from [-10000, 10000] to say [-1000000, 1000000].