Matlab Table Stack and Join Estimation and Simulation Results

back to Fan's Intro Math for Econ, Matlab Examples, or MEconTools Repositories

Combine Tables Together Stack Rows Loop Template Common Columns

There is an estimation routine, each time the routine outputs a table with a single row, the single row contains estimation outputs including estiamtes, standard erros, initial parameters etc. We loop over different estimation routines, with different starting values etc, and rather than saving many tables, we want to save a joint table with all rows stacked together.

This simply means that we have a loop, during each iteration, generating a table, we want to stack things together. For this assume that the column names are the same.

```
for i=1:5
    % a row of coefficent estimates
    rng(123+i);
    it_num_cols = 4;
    it_num_rows = 1;
    mt_saveCoef = rand([it_num_rows, it_num_cols]);

    % row to table
    ar_st_col_names = ["FVAL", "EXITFLAG", "esti_iterations", "esti_funccount"];
    tb_saveCoef = array2table(mt_saveCoef);
    tb_saveCoef.Properties.VariableNames = ar_st_col_names;

    % Stack all results
    if(i == 1)
        tb_saveCoef_stack = tb_saveCoef;
    else
        tb_saveCoef_stack = [tb_saveCoef_stack; tb_saveCoef];
    end
end
% Add esti Counter as column
estimodelctr = (1:size(tb_saveCoef_stack,1))';
tb_saveCoef_stack = addvars(tb_saveCoef_stack, estimodelctr, 'Before', 1);
% Add a row name as a variable
cl_row_names_a = strcat('esti', string((1:size(tb_saveCoef_stack,1))));
tb_saveCoef_stack.Properties.RowNames = cl_row_names_a;
% display results
disp(tb_saveCoef_stack);
```

	estimodelctr	FVAL	EXITFLAG	esti_iterations	esti_funccount
esti1	1	0.10606	0.74547	0.57231	0.45824
esti2	2	0.50673	0.057531	0.62758	0.13255
esti3	3	0.10517	0.12814	0.087406	0.11548
esti4	4	0.52383	0.039963	0.18597	0.77279
esti5	5	0.86664	0.26314	0.13141	0.041593

Combine Tables Together Stack Rows Loop Template Outterjoin

Similar to the previous estimation problem, however, now during different iterations, the column names, i.e. the parameters been estiamted are different. For example, there are 10 parameters, sometimes we estimate 5 of the 10, sometimes 10 or the 10. Want to stack all results together similar to above.

This is accomplished in the following example with the outerjoin function.

```
for i=1:5
    % a row of coefficent estimates
    rng(123+i);

    it_num_rows = 1;
    if (i <= 2)
        it_num_cols = 4;
        mt_saveCoef = rand([it_num_rows, it_num_cols]);
        % row to table
        ar_st_col_names = ["FVAL", "EXITFLAG", "esti_iterations", "esti_funccount"];
    elseif (i <= 4)
        it_num_cols = 2;
        mt_saveCoef = rand([it_num_rows, it_num_cols]);
        % row to table
        ar_st_col_names = ["FVAL", "EXITFLAG"];
    else
        it_num_cols = 3;
        mt_saveCoef = rand([it_num_rows, it_num_cols]);
        % row to table
        ar_st_col_names = ["FVAL", "esti_iterations", "esti_funccount"];
    end

    tb_saveCoef = array2table(mt_saveCoef);
    tb_saveCoef.Properties.VariableNames = ar_st_col_names;

    % Stack all results
    if(i == 1)
        tb_saveCoef_stack = tb_saveCoef;
    else
        tb_saveCoef_stack = outerjoin(tb_saveCoef_stack, tb_saveCoef, 'MergeKeys', true);
    end
end
% Add esti Counter as column
estimodelctr = (1:size(tb_saveCoef_stack,1))';
tb_saveCoef_stack = addvars(tb_saveCoef_stack, estimodelctr, 'Before', 1);
% Add a row name as a variable
cl_row_names_a = strcat('esti', string((1:size(tb_saveCoef_stack,1))));
tb_saveCoef_stack.Properties.RowNames = cl_row_names_a;
% display results
disp(tb_saveCoef_stack);
```

	estimodelctr	FVAL	EXITFLAG	esti_iterations	esti_funccount
esti1	1	0.10517	0.12814	NaN	NaN
esti2	2	0.10606	0.74547	0.57231	0.45824
esti3	3	0.50673	0.057531	0.62758	0.13255
esti4	4	0.52383	0.039963	NaN	NaN

esti5	5	0.86664	NaN	0.26314	0.13141

ND Dimensional Parameter Arrays, Simulate Model and Stack Output Tables

Now we will first column combine matrixes, model parameters and model outcomes, and then row combine matrixes from different simulations.

A model takes a N parameters, solve the model over M sets of parameters. Each time when the model is solved, a P by Q table of results is generated. Each column is a different statistics (mean, std, etc.), and each row is a different outcome variable (consumption, asset choices, etc.). Stack these P by Q Tables together, and add in information about the N parameters, each of the tables been stacked initially had the same column and row names.

The resulting table should have P times M rows, for M sets of model simulations each with P rows of results. And there should be N + Q columns, storing the N parameters as well as the Q columns of different outcomes.

```
rng(123);
% Generate A P by Q matrix of random parameter Values
it_param_groups_m = 5;
it_params_n = 2;
it_outcomes_p = 3;
it_stats_q = 3;

% Parameter Matrix and Names
ar_param_names = strcat('param_', string(1:it_params_n));
mt_param_m_by_n = round(rand([it_param_groups_m, it_params_n])*5, 2);

% Loop over the parameters
for it_cur_param_group=1:1:it_param_groups_m

    % Current Parameters
    ar_param = mt_param_m_by_n(it_cur_param_group,:);

    % Some Model is simulated
    mt_model_simu = normrnd(mean(ar_param), std(ar_param), [it_outcomes_p, it_stats_q]);

    % Model Results are Saved As Table With Column and Row Information
    tb_model_simu = array2table(mt_model_simu);
    cl_col_names = strcat('stats_', string((1:size(mt_model_simu,2))));
    cl_row_names = strcat('outvar_', string((1:size(mt_model_simu,1))));
    tb_model_simu.Properties.VariableNames = cl_col_names;
    tb_model_simu.Properties.RowNames = cl_row_names;

    % Convert Row Variable Names to a Column String
    outvar = string(tb_model_simu.Properties.RowNames);
    tb_model_simu = addvars(tb_model_simu, outvar, 'Before', 1);

    % Parameter Information Table that Shares Row Names as Simu Results
    mt_param_info = zeros([it_outcomes_p,it_params_n]) + ar_param;
    tb_param_info = array2table(mt_param_info);
    tb_param_info.Properties.VariableNames = ar_param_names;
    tb_param_info.Properties.RowNames = cl_row_names;
```

```
    % Combine Parameter Information and Simulation Contents
    tb_model_simu_w_info = [tb_param_info tb_model_simu];
    % Update Row Names based on total row available
    ar_rows_allsimu = (1:it_stats_q)' + (it_cur_param_group-1)*it_stats_q;
    tb_model_simu_w_info.Properties.RowNames = strcat('row=', string(ar_rows_allsimu));

    % Show One Example Table before Stacking
    if (it_cur_param_group == round(it_param_groups_m/2))
        disp(tb_model_simu);
        disp(tb_param_info);
        disp(tb_model_simu_w_info);
    end

    % Stack all results
    if(it_cur_param_group == 1)
        tb_model_allsimu_w_info = tb_model_simu_w_info;
    else
        tb_model_allsimu_w_info = [tb_model_allsimu_w_info; tb_model_simu_w_info];
    end

end
```

	outvar	stats_1	stats_2	stats_3
outvar_1	"outvar_1"	0.056853	2.1703	2.1098
outvar_2	"outvar_2"	3.1545	2.0634	0.7798
outvar_3	"outvar_3"	-0.49033	2.2566	1.7896

	param_1	param_2
outvar_1	1.13	3.42
outvar_2	1.13	3.42
outvar_3	1.13	3.42

	param_1	param_2	outvar	stats_1	stats_2	stats_3
row=7	1.13	3.42	"outvar_1"	0.056853	2.1703	2.1098
row=8	1.13	3.42	"outvar_2"	3.1545	2.0634	0.7798
row=9	1.13	3.42	"outvar_3"	-0.49033	2.2566	1.7896

Show all Simulation Joint Table Outputs:

```
disp(tb_model_allsimu_w_info);
```

	param_1	param_2	outvar	stats_1	stats_2	stats_3
row=1	3.48	2.12	"outvar_1"	2.2665	1.1885	1.924
row=2	3.48	2.12	"outvar_2"	3.3427	2.4647	2.3548
row=3	3.48	2.12	"outvar_3"	2.6714	3.6132	2.918
row=4	1.43	4.9	"outvar_1"	3.3859	5.3759	1.5816
row=5	1.43	4.9	"outvar_2"	3.9499	3.8698	2.2693
row=6	1.43	4.9	"outvar_3"	5.7745	4.6871	1.7334
row=7	1.13	3.42	"outvar_1"	0.056853	2.1703	2.1098
row=8	1.13	3.42	"outvar_2"	3.1545	2.0634	0.7798
row=9	1.13	3.42	"outvar_3"	-0.49033	2.2566	1.7896
row=10	2.76	2.4	"outvar_1"	2.9611	2.6847	2.4986
row=11	2.76	2.4	"outvar_2"	2.9333	2.3457	3.0629
row=12	2.76	2.4	"outvar_3"	2.5814	2.4372	2.4806

row=13	3.6	1.96	"outvar_1"	2.7199	3.3129	3.0577
row=14	3.6	1.96	"outvar_2"	3.9804	1.4529	2.9285
row=15	3.6	1.96	"outvar_3"	2.8445	4.4117	2.6576