

# All Possible Combinations of Arrays as a Table or Random Subset Mesh

back to [Fan's Intro Math for Econ](#), [Matlab Examples](#), or [MEconTools Repositories](#)

## Given Several Arrays, General all Possible Combinations

There are several parameters, might want to simulate at all possible combinations. In the example below, there are four parameters, generate a table with all possible combinations of the four parameters.

```
% A. Quadc linh refh and refsd parameter grids
[it_p1, it_p2, it_p3] = deal(4, 3, 3);
ar_p1 = linspace(-0.09, -0.02, it_p1);
ar_p2 = linspace( 0.020, 0.100, it_p2);
ar_p3 = linspace(-0.100, -0.020, it_p3);
ar_p4 = [0.05];

% B. Mesh Parameters together
% ndgrid mesh together
[mn_p1, ~] = ndgrid(ar_p1, ar_p2, ar_p3, ar_p4);
% combine
[ar_p1_idx, ar_p2_idx, ar_p3_idx, ar_p4_idx] = ind2sub(size(mn_p1), find(mn_p1));
% Index and values
ar_p1_flat = ar_p1(ar_p1_idx)';
ar_p2_flat = ar_p2(ar_p2_idx)';
ar_p3_flat = ar_p3(ar_p3_idx)';
ar_p4_flat = ar_p4(ar_p4_idx)';
mt_paramsmesh_long = [ar_p1_idx(:), ar_p1_flat(:), ...
    ar_p2_idx(:), ar_p2_flat(:), ...
    ar_p3_idx(:), ar_p3_flat(:), ...
    ar_p4_idx(:), ar_p4_flat(:)];
% Sort by a and z
mt_paramsmesh_long = sortrows(mt_paramsmesh_long, [1,3, 5]);

% C. Create Table
tb_paramsmesh_long = array2table(mt_paramsmesh_long);
cl_col_names_a = {'quadc_idx', 'quadc_val', ...
    'linh_idx', 'linh_val', ...
    'refh_idx', 'rehfh_val', ...
    'refsd_idx', 'rehfsd_val'};
tb_paramsmesh_long.Properties.VariableNames = cl_col_names_a;
% D. Display Table
disp(tb_paramsmesh_long);
```

quadc_idx	quadc_val	linh_idx	linh_val	refh_idx	rehfh_val	refsd_idx	rehfsd_val
1	-0.09	1	0.02	1	-0.1	1	0.05
1	-0.09	1	0.02	2	-0.06	1	0.05
1	-0.09	1	0.02	3	-0.02	1	0.05
1	-0.09	2	0.06	1	-0.1	1	0.05
1	-0.09	2	0.06	2	-0.06	1	0.05
1	-0.09	2	0.06	3	-0.02	1	0.05
1	-0.09	3	0.1	1	-0.1	1	0.05

1	-0.09	3	0.1	2	-0.06	1	0.05
1	-0.09	3	0.1	3	-0.02	1	0.05
2	-0.066667	1	0.02	1	-0.1	1	0.05
2	-0.066667	1	0.02	2	-0.06	1	0.05
2	-0.066667	1	0.02	3	-0.02	1	0.05
2	-0.066667	2	0.06	1	-0.1	1	0.05
2	-0.066667	2	0.06	2	-0.06	1	0.05
2	-0.066667	2	0.06	3	-0.02	1	0.05
2	-0.066667	3	0.1	1	-0.1	1	0.05
2	-0.066667	3	0.1	2	-0.06	1	0.05
2	-0.066667	3	0.1	3	-0.02	1	0.05
3	-0.043333	1	0.02	1	-0.1	1	0.05
3	-0.043333	1	0.02	2	-0.06	1	0.05
3	-0.043333	1	0.02	3	-0.02	1	0.05
3	-0.043333	2	0.06	1	-0.1	1	0.05
3	-0.043333	2	0.06	2	-0.06	1	0.05
3	-0.043333	2	0.06	3	-0.02	1	0.05
3	-0.043333	3	0.1	1	-0.1	1	0.05
3	-0.043333	3	0.1	2	-0.06	1	0.05
3	-0.043333	3	0.1	3	-0.02	1	0.05
4	-0.02	1	0.02	1	-0.1	1	0.05
4	-0.02	1	0.02	2	-0.06	1	0.05
4	-0.02	1	0.02	3	-0.02	1	0.05
4	-0.02	2	0.06	1	-0.1	1	0.05
4	-0.02	2	0.06	2	-0.06	1	0.05
4	-0.02	2	0.06	3	-0.02	1	0.05
4	-0.02	3	0.1	1	-0.1	1	0.05
4	-0.02	3	0.1	2	-0.06	1	0.05
4	-0.02	3	0.1	3	-0.02	1	0.05

## Matlab Draw Random with and without Replacement

```
%Generate a matrix named foo, with limited numbers
rng(1234);
foo = unique((round((randn(5,1)+1)*100)));
disp(foo);
```

```
5
78
154
219
232
```

```
% draw 10 random samples without replacement
index = randsample(1:length(foo), 4);
bar_rand_noreplace = foo(index,:);

% draw 1000 random samples with replacement
index = randsample(1:length(foo), 4, true);
bar_rand_replace = foo(index,:);

% Display
disp(table(bar_rand_noreplace, bar_rand_replace));
```

bar_rand_noreplace	bar_rand_replace
5	78
78	154

## Matrix Meshgrid to Loop Permutated Vectors

Meshgrid to generate all permutations of arrays.

```
k = linspace(1,10,10);
kp = linspace(1,10,10);
z = linspace(0,1,10);

[kM kpM zM] = meshgrid(k,kp,z);
kMVec = kM(:);
kMpVec = kpM(:);
zMVec = zM(:);

outputVec = zeros(size(zMVec));
for a=1:length(zMVec)
    outputVec(a) = kMVec(a)+kMpVec(a)+zMVec(a);
end

outputTens = reshape(outputVec,size(kM));
disp(outputTens);
```

(:,:,1) =

2	3	4	5	6	7	8	9	10	11
3	4	5	6	7	8	9	10	11	12
4	5	6	7	8	9	10	11	12	13
5	6	7	8	9	10	11	12	13	14
6	7	8	9	10	11	12	13	14	15
7	8	9	10	11	12	13	14	15	16
8	9	10	11	12	13	14	15	16	17
9	10	11	12	13	14	15	16	17	18
10	11	12	13	14	15	16	17	18	19
11	12	13	14	15	16	17	18	19	20

(:,:,2) =

2.1111	3.1111	4.1111	5.1111	6.1111	7.1111	8.1111	9.1111	10.1111	11.1111
3.1111	4.1111	5.1111	6.1111	7.1111	8.1111	9.1111	10.1111	11.1111	12.1111
4.1111	5.1111	6.1111	7.1111	8.1111	9.1111	10.1111	11.1111	12.1111	13.1111
5.1111	6.1111	7.1111	8.1111	9.1111	10.1111	11.1111	12.1111	13.1111	14.1111
6.1111	7.1111	8.1111	9.1111	10.1111	11.1111	12.1111	13.1111	14.1111	15.1111
7.1111	8.1111	9.1111	10.1111	11.1111	12.1111	13.1111	14.1111	15.1111	16.1111
8.1111	9.1111	10.1111	11.1111	12.1111	13.1111	14.1111	15.1111	16.1111	17.1111
9.1111	10.1111	11.1111	12.1111	13.1111	14.1111	15.1111	16.1111	17.1111	18.1111
10.1111	11.1111	12.1111	13.1111	14.1111	15.1111	16.1111	17.1111	18.1111	19.1111
11.1111	12.1111	13.1111	14.1111	15.1111	16.1111	17.1111	18.1111	19.1111	20.1111

(:,:,3) =

2.2222	3.2222	4.2222	5.2222	6.2222	7.2222	8.2222	9.2222	10.2222	11.2222
3.2222	4.2222	5.2222	6.2222	7.2222	8.2222	9.2222	10.2222	11.2222	12.2222
4.2222	5.2222	6.2222	7.2222	8.2222	9.2222	10.2222	11.2222	12.2222	13.2222
5.2222	6.2222	7.2222	8.2222	9.2222	10.2222	11.2222	12.2222	13.2222	14.2222
6.2222	7.2222	8.2222	9.2222	10.2222	11.2222	12.2222	13.2222	14.2222	15.2222
7.2222	8.2222	9.2222	10.2222	11.2222	12.2222	13.2222	14.2222	15.2222	16.2222



3.7778	4.7778	5.7778	6.7778	7.7778	8.7778	9.7778	10.7778	11.7778	12.7778
4.7778	5.7778	6.7778	7.7778	8.7778	9.7778	10.7778	11.7778	12.7778	13.7778
5.7778	6.7778	7.7778	8.7778	9.7778	10.7778	11.7778	12.7778	13.7778	14.7778
6.7778	7.7778	8.7778	9.7778	10.7778	11.7778	12.7778	13.7778	14.7778	15.7778
7.7778	8.7778	9.7778	10.7778	11.7778	12.7778	13.7778	14.7778	15.7778	16.7778
8.7778	9.7778	10.7778	11.7778	12.7778	13.7778	14.7778	15.7778	16.7778	17.7778
9.7778	10.7778	11.7778	12.7778	13.7778	14.7778	15.7778	16.7778	17.7778	18.7778
10.7778	11.7778	12.7778	13.7778	14.7778	15.7778	16.7778	17.7778	18.7778	19.7778
11.7778	12.7778	13.7778	14.7778	15.7778	16.7778	17.7778	18.7778	19.7778	20.7778

(:,:,9) =

2.8889	3.8889	4.8889	5.8889	6.8889	7.8889	8.8889	9.8889	10.8889	11.8889
3.8889	4.8889	5.8889	6.8889	7.8889	8.8889	9.8889	10.8889	11.8889	12.8889
4.8889	5.8889	6.8889	7.8889	8.8889	9.8889	10.8889	11.8889	12.8889	13.8889
5.8889	6.8889	7.8889	8.8889	9.8889	10.8889	11.8889	12.8889	13.8889	14.8889
6.8889	7.8889	8.8889	9.8889	10.8889	11.8889	12.8889	13.8889	14.8889	15.8889
7.8889	8.8889	9.8889	10.8889	11.8889	12.8889	13.8889	14.8889	15.8889	16.8889
8.8889	9.8889	10.8889	11.8889	12.8889	13.8889	14.8889	15.8889	16.8889	17.8889
9.8889	10.8889	11.8889	12.8889	13.8889	14.8889	15.8889	16.8889	17.8889	18.8889
10.8889	11.8889	12.8889	13.8889	14.8889	15.8889	16.8889	17.8889	18.8889	19.8889
11.8889	12.8889	13.8889	14.8889	15.8889	16.8889	17.8889	18.8889	19.8889	20.8889

(:,:,10) =

3	4	5	6	7	8	9	10	11	12
4	5	6	7	8	9	10	11	12	13
5	6	7	8	9	10	11	12	13	14
6	7	8	9	10	11	12	13	14	15
7	8	9	10	11	12	13	14	15	16
8	9	10	11	12	13	14	15	16	17
9	10	11	12	13	14	15	16	17	18
10	11	12	13	14	15	16	17	18	19
11	12	13	14	15	16	17	18	19	20
12	13	14	15	16	17	18	19	20	21

## Given Integer Arrays, All Possible Combinations

given any sizes arrays, N of them, create all possible combinations

```
ar_it_a = 1:3;
ar_it_b = 1:2;
ar_it_c = 2:4;
ar_it_d = -1:-1:-2;
ar_it_e = 0.1;

cl_ar_all = {ar_it_a, ar_it_b, ar_it_c, ar_it_d, ar_it_e};
cl_mt_all = cl_ar_all;
[cl_mt_all{:}] = ndgrid(cl_ar_all{:});
mt_it_allcombo = cell2mat(cellfun(@(m) m(:), cl_mt_all, 'uni', 0));

disp(mt_it_allcombo)
```

1.0000	1.0000	2.0000	-1.0000	0.1000
2.0000	1.0000	2.0000	-1.0000	0.1000
3.0000	1.0000	2.0000	-1.0000	0.1000
1.0000	2.0000	2.0000	-1.0000	0.1000
2.0000	2.0000	2.0000	-1.0000	0.1000
3.0000	2.0000	2.0000	-1.0000	0.1000

1.0000	1.0000	3.0000	-1.0000	0.1000
2.0000	1.0000	3.0000	-1.0000	0.1000
3.0000	1.0000	3.0000	-1.0000	0.1000
1.0000	2.0000	3.0000	-1.0000	0.1000
2.0000	2.0000	3.0000	-1.0000	0.1000
3.0000	2.0000	3.0000	-1.0000	0.1000
1.0000	1.0000	4.0000	-1.0000	0.1000
2.0000	1.0000	4.0000	-1.0000	0.1000
3.0000	1.0000	4.0000	-1.0000	0.1000
1.0000	2.0000	4.0000	-1.0000	0.1000
2.0000	2.0000	4.0000	-1.0000	0.1000
3.0000	2.0000	4.0000	-1.0000	0.1000
1.0000	1.0000	2.0000	-2.0000	0.1000
2.0000	1.0000	2.0000	-2.0000	0.1000
3.0000	1.0000	2.0000	-2.0000	0.1000
1.0000	2.0000	2.0000	-2.0000	0.1000
2.0000	2.0000	2.0000	-2.0000	0.1000
3.0000	2.0000	2.0000	-2.0000	0.1000
1.0000	1.0000	3.0000	-2.0000	0.1000
2.0000	1.0000	3.0000	-2.0000	0.1000
3.0000	1.0000	3.0000	-2.0000	0.1000
1.0000	2.0000	3.0000	-2.0000	0.1000
2.0000	2.0000	3.0000	-2.0000	0.1000
3.0000	2.0000	3.0000	-2.0000	0.1000
1.0000	1.0000	4.0000	-2.0000	0.1000
2.0000	1.0000	4.0000	-2.0000	0.1000
3.0000	1.0000	4.0000	-2.0000	0.1000
1.0000	2.0000	4.0000	-2.0000	0.1000
2.0000	2.0000	4.0000	-2.0000	0.1000
3.0000	2.0000	4.0000	-2.0000	0.1000