Maplab Nested Container Map with Struct

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

A Struct of Model Parameters with Different Information

There is a list of model parameters, there are various information we store for each parameter. Store each type of information in a different container map, and then combine them together in a struct. This is more flexible than generating a table, and can be called with a single line. This is effectively a nested container, imagine if we define for each parameter a map with keys indicating different types of information. Rather than doing that, the keys are elements of the struct, each key/value is in a different container.

```
% index for different parameters
mp_param_idx = containers.Map('KeyType','char', 'ValueType','any');
mp_param_idx('NPquad') = 3;
mp_param_idx('gamma') = 4;
mp_param_idx('HAquad') = 5;
mp_param_idx('theta') = 6;
mp_param_idx('lambda') = 7;
mp_param_idx('msrErrProtSD') = 8;
mp_param_idx('logProt') = 9;
mp_param_idx('freePriceFrac') = 10;
mp_param_idx('h_exoshk_sd') = 11;
mp_param_idx('h_endoshk_sd') = 12;

% Invert key and index
mp_idx_params = containers.Map(...
    cellfun(@(idx) num2str(idx(:)), values(mp_param_idx), 'uni', 0), ...
    keys(mp_param_idx));

% Exponentiation Positivity Restrictions
mp_param_explog = containers.Map('KeyType','char', 'ValueType','any');
mp_param_explog('NPquad') = -1;
mp_param_explog('gamma') = 1;
mp_param_explog('lambda') = -1;
mp_param_explog('msrErrProtSD') = 1;
mp_param_explog('freePriceFrac') = 1;
mp_param_explog('h_exoshk_sd') = 1;
mp_param_explog('h_endoshk_sd') = 1;

% Create Struct
param_struct.paramindex = mp_param_idx;
param_struct.paramstring = mp_idx_params;
param_struct.explog = mp_param_explog;
```

Given the struct contructed, can get the index for a particular parameter, or the explog value in a single line call:

```
% Get values in single line
disp(['NPquad index=' num2str(param_struct.index('NPquad')) ...
    ', explog=' num2str(param_struct.explog('NPquad'))]);
```

```
NPquad index=3, explog=-1
```

Furthermore, since we have both mp_param_idx and mp_idx_params, suppose we only know the current index, we can use to index to find the string, and use the string to find the expolog value:

```
% Get the explog value for a particular index
st_param = param_struct.paramstring('11');
it_explog = param_struct.explog(st_param);
% Single line call
disp(['The explog of parameter index 11 is "' ...
    num2str(param_struct.explog(param_struct.paramstring('11'))) ...
    '" (param index 11 is "' st_param '")']);
```

```
The explog of parameter index 11 is "1" (param index 11 is "h_exoshk_sd")
```