Equilibrium function

Table of Contents

- Filename: equil.m
- Authors: Matt Reimer
- Created: 07/29/17
- Purpose: Function that returns the value of the equilibrium condition. This function will be used to find the vector of collocation nodes such that this output from this function is equal to zero.

Description

The function `equil` returns the value of the equilibrium condition at all collocation nodes. The ultimate goal is to find the matrix of collocation coefficients such that the output from the function is equal to zero, which gives us our REE quota price function.

Specifically, let $\mathbf{I}_t$ be a d-dimensional row vector, where d denotes the number of state variables, $\mathbf{w}\left(\mathbf{I}_t;\eta\right)$ be an S-dimensional row vector of expected quota prices given the information in $\mathbf{I}_t$, and $\mathbf{e}\left(\mathbf{w}\left(\mathbf{I}_t;\eta\right)\right)$ be an S-dimensional row vector of end-of-season excess demand for quota. Then for an RE equilibirum to exist, we must have that:

$$\mathbf{F}\left(\mathbf{w}\left(\mathbf{I}_t;\eta\right)\right) = \max\left\{\mathbf{e}\left(\mathbf{w}\left(\mathbf{I}_t;\eta\right)\right), -\mathbf{w}\left(\mathbf{I}_t;\eta\right)\right\} = \mathbf{0}.$$

This function evaluates $\mathbf{F}$ at N collocation nodes, and returns a column vector of length $S \times N$.

```
function F = equil(eta,I,m)
```

Input arguments:

- `eta` = a $NS \times 1$ vector of collocation coefficients;
- `I` = a $NS \times d$ matrix of collocation nodes;
- `m` = a structural array containing parameter values

Output arguments:

- `F` = a $NS \times 1$ vector of function values.

Notes:

The vector η will be provided by a Matlab solver (e.g., `fsolve`).

Preliminaries

```
n = prod(m.fspace.n);               % Total # of nodes to evaluate
eta = reshape(eta,n,m.model.S);     % Transform into matrix--each
column represents a quota price
```

Calculate Excess Demand and End-of-Season Quota Prices

```
xdem = zeros(n,m.model.S);
wend = zeros(n,m.model.S);
for i = 1:n
    [xdem(i,:),wend(i,:)] = xdemand(eta,I(i,:),m);
end
```

See if Equilibirum Condition is Satisfied

```
F = max(xdem(:),-wend(:));
% If minimizing sum of squared residuals
%F = sum(F.^2);

end
```

Published with MATLAB® R2016b