
```

function [PI, Orig, Lambda] = myGenetic(f, interval, parents, TOL_GA,
    G, S, dv)
% Input parameters
% parents: scalar, number of design strings to preserve and breed
% TOL_GA: scalar, acceptable cost function threshold to stop evolution
% G: scalar, maximum total generations
% S: scalar, number of total design strings per generation
% dv: scalar, number of design variables per string
% f: a function that can take in either a scalar or a column vector of
% design parameters
% Pretty sure there should be a search interval specified here

% Outputs
% PI: G x S array where PI(g,s) is the cost of the s'th ranked design
% in the g'th generation
% Orig: G x S array where Orig(g,:) represents the indices of each
% sorted entry from before they were sorted.
% Lambda: S x dv array of the most recent generations design strings

if mod(parents,2) ~= 0
    error('Number of parents must be even.')
end
if parents > S
    error('Parents to keep must be less than or equal to population
        size')
end

K = parents/2;
pop = randi(100*interval, [dv, S])/100; % Creates a dv x S matrix
    where each column is a design
raw_cost = zeros(1,S); % Preallocate generation cost vector
best_parents = zeros(dv, parents);

for g = 1:G

    for s = 1:S
        func_vals = num2cell(pop(:,s));
        raw_cost(s) = f(func_vals{:}); % Calculate cost of each
design
    end
    [sorted_cost, sorted_indices] = sort(raw_cost, 2);
    PI(g,:) = sorted_cost;
    Orig(g,:) = sorted_indices;

    best_indices = sorted_indices(1:parents);

```

```

        best_parents(:, 1:parents) = pop(:, best_indices); % each column
        is a parent. each row is lambda_n
        children = zeros(dv, parents); % preallocate children array
        for k = 2*(1:K)
            for i = 1:-1:0
                phi = rand(1);
                child = best_parents(:,k-1).*phi + best_parents(:,k).*(1-
phi);
                children(:,k-i) = child;
            end

        end

        pop = [best_parents, children, randi(100*interval, [dv,
S-2*parents])/100]; % Creates a dv x S matrix where each column is a
design

        if sorted_cost(1)<TOL_GA
            break
        end
    end

    Lambda = pop';

end

Not enough input arguments.

Error in myGenetic (line 20)
if mod(parents,2) ~= 0

```

Published with MATLAB® R2019a