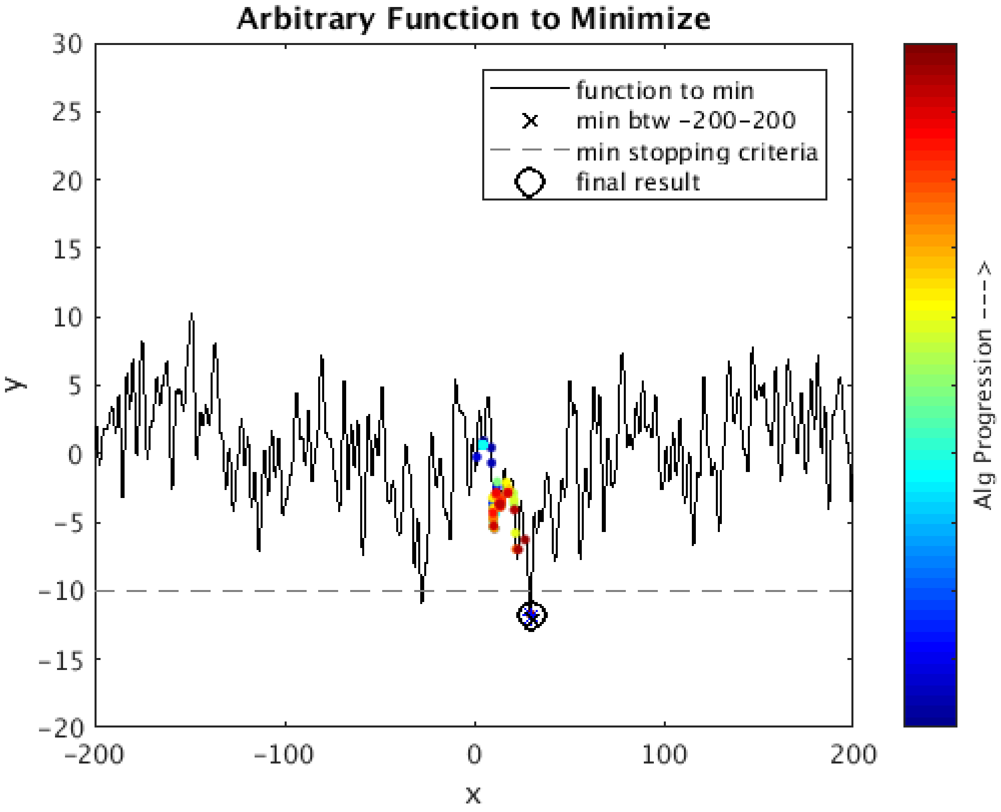
```
% Paula Burgi
% HW 7, prob 2

close all;
clear;


% make complicated function
x = sym('x');
s = sin(x);
for i = 1:20
    a = rand(1);
    b = randn(1);
    c = randn(1);
    d = rand(1);
    s = s + a.*sin(b*x) + c.*cos(d*x);
end
s = s+(abs(x)./30)-3;
x     = -200:0.1:200;
xa    = x;
ss    = eval(s);
[m,d] = min(ss);

% guess
x = -50;
xo = x;
eo = eval(s);
nT = 1e4;
nits = 100;


% stoping criteria
eStop = -10;

% initiate plot
figure; hold on; box on;
plot(xa, ss, 'k');
plot(xa(d),m, 'kx', 'linewidth', 2);
plot(xa, ones(1, length(xa)).*eStop, '--', 'color', [0.5 0.5 0.5]);
xlabel('x');
ylabel('y');
title('Arbitrary Function to Minimize');
ylim([-20 30]);

% intital candidate individual (composed of sum of numbers)
% population is make of individuals
p = [];
inds = [];
np = 10;
for i = 1:np
    gi = rand(10, 1);
    indi = sum(gi);
    inds = [inds; gi];
    p = [p indi];
end
p1 = p;

% start iterations
eia = [];
```

```
xoa = [];
for i = 1:nits
    % fitnesses (determined by cost function)
    ei = []; % value of function at position of each individual
    for j = 1:np
        x = p(j);
        ei = [ei; eval(s)];
    end
    [eis, ids] = sort(ei, 'ascend');
    rr = (eis(end)-eis)+.01;
    % fitness defined as normalized difference between each individual to worst
    % individual (plus small constant)
    f1  = rr(ids)/sum(rr);
    f = [0; cumsum(f1)];
    pf = p;
    [eo, mid] = min(ei);
    eia = [eia; eo];
    xo  = pf(mid);
    xoa = [xoa; xo];

    % stopping criteria
    if eo < eStop
        break
    end


    % selection: roulette wheel
    % surviving individuals
    r = rand (1000,1);
    idx = discretize(r,f);
    sinds = unique(idx, 'stable');
    sinds = sinds(1:5); % choose first 5 unique results of roulette wheel

    % crossover
    pn = p(sinds);
    en = ei(sinds);
    % crossover involves adding fraction of x to y
    fr = 0.5;
    p = [pn+(1-fr)*en; pn-(1-fr)*en];

    % mutation
    % introduce random mutation based on outcome of random probablility
    % (smaller than 5%)
    for j = 1:np
        mutr = rand;
        if mutr < 0.05;
            p(j) = p(j)+rand.*5;
        end
    end

end

plot(xo, eo, 'ko', 'markersize', 10, 'linewidth', 2);
ll = length(eia);
cm = jet(ll);
for i=1:ll
    plot(xoa(i), eia(i), '.', 'color', cm(i,:), 'markersize', 10);
end
plot(xo, eo, 'b*');
plot(xa(d),m, 'kx', 'linewidth', 2, 'linewidth', 2);
colormap jet
```

```
h = colorbar;
set(h, 'TickLabels', ['']);
h.Label.String = 'Alg Progression --->';
legend('function to min', 'min btw -200-200', 'min stopping criteria', 'final result');
```



Published with MATLAB® R2017a