```matlab
function p_current = NewtonsMethod2()
% Use Newton's method to approximate the zero of the function
% f(x) = x^2 - 2xe^(-x) + e^(-2x) to within 10^(-8)
% with p0 = 0
% Return the value of pn for the approximation

% how many iterations it takes to approximate the zero of the function
how_many_iterations = 0;

% tolerance
TOL = 10^(-8);

% pn-1
p_last = 0;

% pn
p_current = 1;

% f(x) = x^2 - 2xe^(-x) + e^(-2x)
f  = @(x) x^2 -2*x*exp(-x) + exp(-2*x);

% the derivative of f(x)
df = @(x) 2*x -2*exp(-x);

% Until the error range is less than tolerance,
% we continue to apply Newton's method.
while abs((p_current - p_last)/p_current) >= TOL

    % set pn-1 to the last value of pn
    p_last = p_current;

    % calculate pn = pn-1 - f(pn-1)/f'(pn-1)
    p_current = p_last - f(p_last) / df(p_last);

    % increment how_many_iterations
    how_many_iterations = how_many_iterations + 1;

end

% print the results
fprintf("n: %d\t", how_many_iterations);
fprintf("p%d:  %.8f\t", how_many_iterations, p_current);
fprintf("|error|: %.8f\n", TOL);

end

n: 13 p13:  0.56714329 |error|: 0.00000001

ans =

    0.5671
```

Published with MATLAB® R2018b