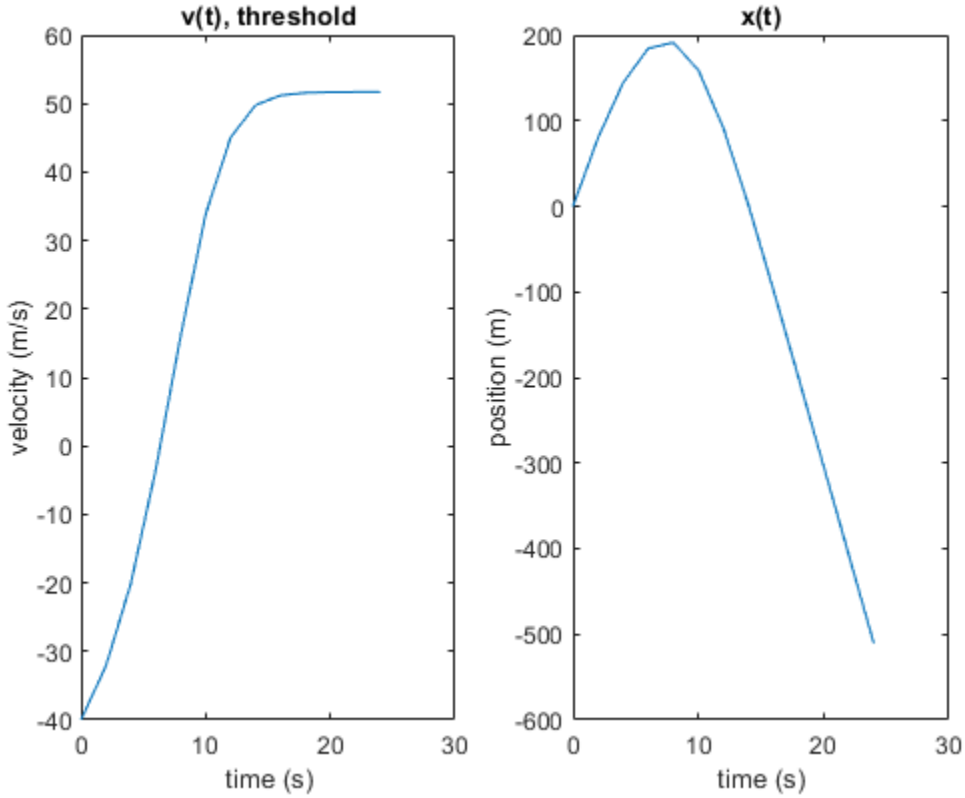
Numerical Solution to the Bungee Jumper Problem

Note: Incorrect Drag Model

```
% clear;

a_y_init = 9.81; % m/s^2
mass = 68.1; % kg
c_d = .25; % kg/m;

t_init = 0; % s
t_delta = 2; % s
t_current = t_init; % s
t_vector = []; % s

v_init = -40; % m/s
v_previous = v_init; % m/s
v_current = v_init; % m/s
v_threshold = .01; % m/s
v_time = [];

x_init = 0; % m
x_current = x_init; % m
x_previous = x_init;
x_time = []; % m

iteration = 1;
done = false;

t_vector( iteration ) = t_init;
v_time( iteration ) = v_current;

while (done == false)
    t_current = t_current + t_delta;
    v_current = v_previous + ( (a_y_init - (( c_d / mass ) *
 ( v_previous ^ 2))) * t_delta );
    x_current = x_previous - ( v_previous * t_delta );

    if (abs(v_current - v_previous) > v_threshold )
        % keep on calculting
    else
        done = true;
    end
    iteration = iteration + 1;
    t_vector( iteration ) = t_current;

    v_time( iteration ) = v_current;
    v_previous = v_current;

    x_time( iteration ) = x_current;
```

```
    x_previous = x_current;
end

subplot( 1,2,1 )
plot( t_vector, v_time )
title('v(t), threshold');
xlabel('time (s)');
ylabel('velocity (m/s)');

subplot( 1,2,2 )
plot( t_vector, x_time )
title('x(t)');
xlabel('time (s)');
ylabel('position (m)');
```



Published with MATLAB® R2018a