```
% Dubins_DataGenScript.m is used to generate training data to be used for a
% machine learning algorithm that classifies reachability sets.
%
%   Ross Allen, ASL, Stanford University
%   Oct 24, 2013
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

clear
clc
close all

m = 1000;                       % number of training examples
n = 3;                         % dimension of attribute variable
numerics.n_nodes = 30;         % number of discretization nodes

% extents of attributes to be tested
delx_extent = [-10 10];         %(m)  delta-x
dely_extent = [-10 10];         %(m)  delta-y
deltheta_extent = [-360 360];   %(deg)delta-theta
% rho_range = [0.1 10];          %(m)  minimum turning radius
V = 1;                         %(m/s) velocity
turnrate = 45;                 %(deg/s) maximum turning rate
extents = [delx_extent; dely_extent; deltheta_extent];

% Generate training data input values
halset = haltonset(n);            % halton set to sample input ranges
halset = scramble(halset, 'RR2'); % scramble the set with reverse-rad
halset = net(halset, m);   % extract first m samples of scrambled set
inputset = halset*diag([-1 1]*extents') + ones(m,n)*diag(extents(:,1));

% Environment
environment.xbounds = [-inf inf]; % (m, m)
environment.ybounds = [-inf inf]; % (m, m)

% Set solver options
options.print_summary = false; % (boo)
options.plot_results = false;  % (b00)
options.solver = optimset('Algorithm','sqp','GradObj','on',...
    'GradConstr','on','DerivativeCheck','off', 'Display', 'off',...
    'TolFun', 1e-6, 'TolCon', 1e-6, 'TolX', 1e-6);
options.solver.MaxFunEvals = Inf;
options.solver.MaxIter = 2000;

% Solve optimal control problem
outputset = NaN*ones(m,1);
exitflags = NaN*ones(m,1);
for i = 1:m
    i

    % Set robot values
    robot.V = V;
    robot.turnrate = turnrate;

    % Set bounday values
    bv = inputset(i,:);
    boundary_values.x0 = 0;
    boundary_values.y0 = 0;
    boundary_values.theta0 = 0;
    boundary_values.t0 = 0;
    boundary_values.xf = bv(1);
    boundary_values.yf = bv(2);
    boundary_values.thetaf = bv(3);

    % Consolidate
    dubprob.numerics = numerics;
    dubprob.robot = robot;
    dubprob.boundary_values = boundary_values;
    dubprob.environment = environment;
    dubprob.options = options;
    clear robot boundary_values

    % Call Solver
    dubprob = DubinsOptimizer(dubprob);

    % Save optimal cost and exit flag
    exitflags(i) = dubprob.solution.exitflag;
    outputset(i) = dubprob.solution.cost;

    clear dubprob
end

% Save Data
trainingdata = [(1:m)' inputset outputset exitflags];

%% Write to file
filename = 'DubinsTrainData-Oct26-2.txt';
dlmwrite(filename, trainingdata,'delimiter',';','precision', 5,  'newline', 'pc')
```