```
function [weight_matrix accuracy] = linear_MC_optimizer(dataset, capacity,
simulation_depth, N)

  num_rows = size(dataset, 1);
  weight_matrix = zeros(simulation_depth, N); %this is an initial seed value for
  the Monte Carlo algorithm

  %this ensures the value of alpha is 1 for the first iteration------------------
  accuracy = zeros(simulation_depth, 1);

  class_vector = dataset(:, N + 1);

  for j = 1 : simulation_depth

    alpha = (1 - accuracy(j))/(1 + accuracy(j))^(1 + accuracy(j));
    initial_weight_vector = weight_matrix(j,:);

    [predicted_class_vector diff_vector weight_vector] =
    interp_classification(dataset, capacity, initial_weight_vector, alpha, N);

    weight_matrix(j + 1,:) = weight_vector; %these are the weights generated

    num_errors = sum(predicted_class_vector != class_vector);
    accuracy(j + 1) = 1 - num_errors/num_rows;

  endfor

endfunction
```