```
function [x y z s c] = imitate_data(anchor_tree, delta_tree, predict_z_toggle,
num_data_points, x_domain, y_domain, neg)

  num_iterations = floor(num_data_points/25);

  cnt = 1;

for i = 1 : num_iterations;

x_base = i*(x_domain / num_iterations);
y_base = i*(y_domain / num_iterations);

  for j = 1 : 25

    if(neg == 1)

      x(cnt) = x_base*(rand()*2 - 1);
      y(cnt) = y_base*(rand()*2 - 1);
      z(cnt) = 1*rand();

    else

      x(cnt) = x_base*rand();
      y(cnt) = y_base*rand();
      z(cnt) = 1*rand();

    endif

    new_data_item{1} = [x(cnt) y(cnt) z(cnt)];

    s(cnt) = 50;

    [category_index predicted_z final_delta] = predict_best_fit_tree(anchor_tree,
delta_tree, new_data_item,predict_z_toggle);

    if(predicted_z != Inf)

      z(cnt) = predicted_z;

      c(cnt,1) = .01;
      c(cnt,2) = .4;
      c(cnt,3) = .9;

    else

      z(cnt) = 0;

      c(cnt,1) = .8;
```

```
      c(cnt,2) = .8;
      c(cnt,3) = .8;

    endif

    cnt = cnt + 1;

  endfor

endfor

endfunction
```