

```

%=====
%=====
%VECTORIZED DEEP LEARNING
%=====
%=====
%COPYRIGHT CHARLES DAVI 2020

function [final_dataset final_weight_vector final_nearest_neighbors] = normalize_datasetBT(dataset,N)
%=====
%NORMALIZES DATASET
%=====

num_rows = size(dataset,1);
%calculate the average number of digits in each dimension-----
for i = 1 : N

    digit_vector(i) = spec_logBT(mean(abs(dataset(:,i))))/log2(10);

endfor

%Note that N+1 is never included
max_dim = ceil(max(digit_vector));
min_dim = floor(min(digit_vector));

max_dimensions_vector = 1:N;

%normalizes the largest category of dimensions-----
num_items = size(max_dimensions_vector,2);
weight_vector = ones(1,N+1); %used to normalize the dataset
final_dataset = dataset;
final_weight_vector = weight_vector;

%sets the original accuracy
nearest_neighbors = NN_fully_vectorized(dataset, N);
actual_class_vector = dataset(:,N+1);
predicted_class_vector = dataset(nearest_neighbors, N+1);
final_nearest_neighbors = nearest_neighbors;

%calculates the error
error_vector = predicted_class_vector != actual_class_vector;
num_errors = sum(error_vector);
accuracy = 1 - num_errors/num_rows;
max_accuracy = accuracy;

%tests different weights for each dimension in scope
for i = min_dim : max_dim

    for j = 1 : num_items

        current_index = max_dimensions_vector(j);
        current_num_digits = digit_vector(current_index);

        diff = i - current_num_digits;

        weight_vector(current_index) = 10^diff;

    endfor

    modified_dataset = dataset.*weight_vector;
    nearest_neighbors = NN_fully_vectorized(modified_dataset,N);

```

```
actual_class_vector = dataset(:,N+1);
predicted_class_vector = dataset(nearest_neighbors, N+1);

%calculates the error
error_vector = predicted_class_vector != actual_class_vector;
num_errors = sum(error_vector);
accuracy = 1 - num_errors/num_rows;

if(accuracy > max_accuracy)

    max_accuracy = accuracy;
    final_weight_vector = weight_vector;
    final_dataset = modified_dataset;
    final_nearest_neighbors = nearest_neighbors;

endif

endfor

endfunction
```