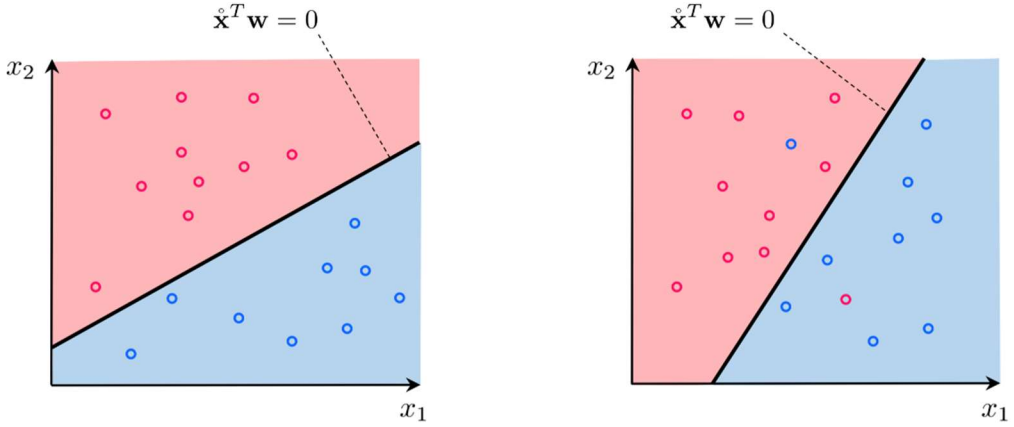
ECEC 487 – Homework 2 Outputs/write up

Problem 1

Continuous piecewise linear functions are smooth function with only a few 'bumps'. They are examples of piecewise differentiable functions. They're built from different pieces of the function within certain ranges and differentiable within these ranges. As we can see from the graph below the cost function graph is a continuous piecewise linear function because it's smooth and differentiable along the line $X^T w$

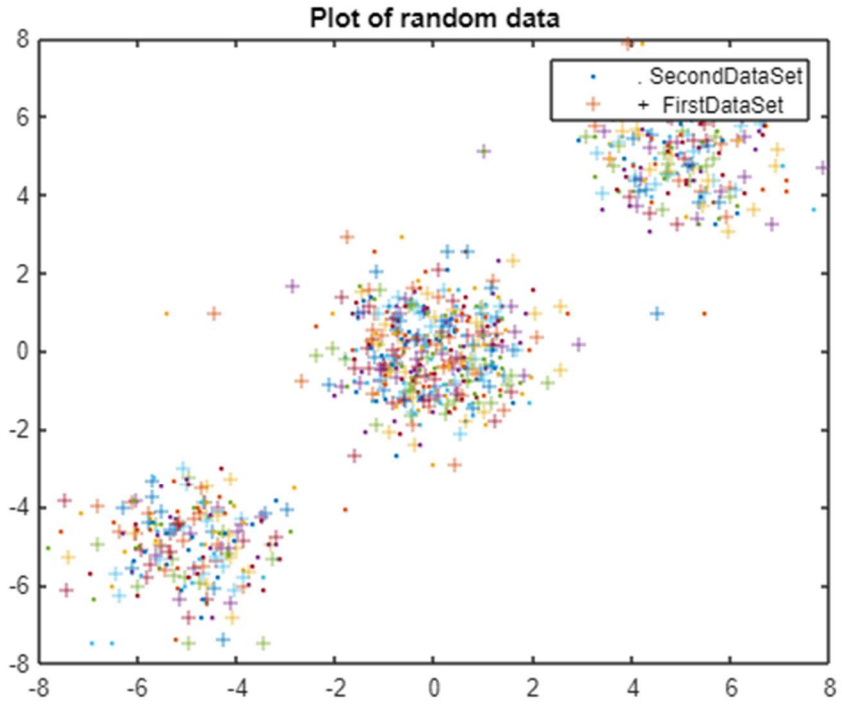


Reference for figure: Machine Learning Refined, Jeremy Watt, second ed, 2020

Experiment 1

```
FirstDataSet = 2×202
     -3.8350   -4.3732   -4.9249   -4.6484   -5.6965   -3.3039   -4.9409   -3.2029 ...
     -0.1274    0.5542   -1.0973   -0.7313    1.4047   -0.6202    0.2371   -1.5868

SecondDataSet = 2×202
     -5.7027   -4.6436   -4.3474   -4.7843   -5.2639   -3.1976   -5.6430   -4.8904 ...
      0.4185    0.2473    0.7041    0.6319   -0.9924    1.7667   -0.3821   -0.9114



w_perceptron_FirstDataSet = 2×1
      1.2656
     -1.2612

iter = 20000
mis_clas = 1
w_perceptron_SecondDataSet = 2×1
      1.2457
     -1.2441

iter = 20000
mis_clas = 1
w_SSErr_FirstDataSet = 2×1
      0.1928
     -0.0013

w_SSErr_SecondDataSet = 2×1
      0.1932
     -0.0198

w_LMS_FirstDataSet = 2×1
      0.1976
      0.5428

w_LMS_SecondDataSet = 2×1
      0.1942
      0.6760

performance_perceptron_FrstDataSet = 0.4511
performance_perceptron_SecondDataSet = 0.4387
performance_SSErr_FrstDataSet = 7.3192e-06
```

```
performance_SSErr_SecondDataSet = 7.0301e-06
performance_LMS_FirstDataSet = 9.6119e-05
performance_LMS_SecondDataSet = 5.2525e-05
```




```
Error using plot_NN_reg (line 34)
```

Error plotting decision boundary

I started off the experiment by generating two random variables X1 and X'^1 using the 'mvnrnd' function in Matlab. The first half and second half of both data sets had different mean vectors, so I had to split the 'mvnrnd' function into two for each data set, appended a 1 and then concatenated them.

Next, I used the 'perce', 'LMSalg', and 'SSErr' functions to apply the perceptron and LMS algorithms, and Sum of error classifiers, respectively, on both data sets I generated.

After that, I measured the performance value of the functions by using the 'timeit' function and we can see that SSErr is the fastest function to find the parameter vector for both data sets.

Finally, I plotted the graph for X1, X'^1, and the line corresponding to our parameter vector by using