Table of Contents

Open ended 3.1

```
%{
 The Perceptron algo is a continuous piecewise linear function because
 it must stay within the realm of real numbers and must seperate the
 two
 classes indefinitely. It must be real as the wights mapped by w are
 real,
 and since the weights define the equation of the line it must be
 real.
%}
```

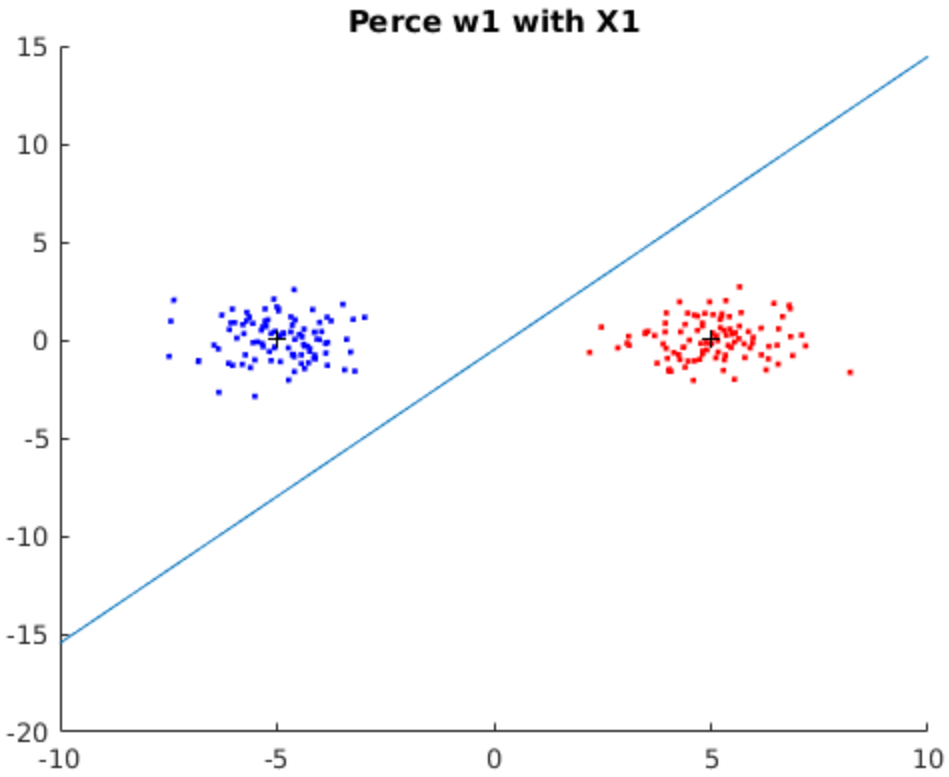
Problem 3.1

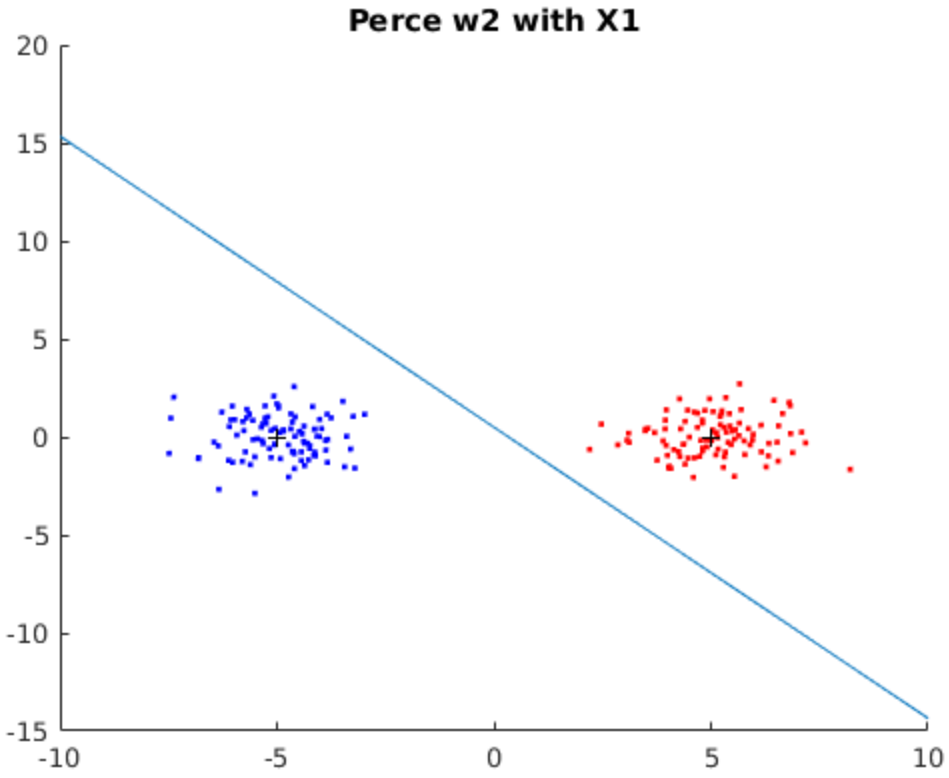
```
clear all
clc
randn('seed', 0);

N = 200;
m = [-5 5 ;0 0];
Si = [];
P = [1/2, 1/2];
for i = 1:2
 Si(:,:,i) = [1 0; 0 1];
end


[X1, y1] = genGaussClasses(m, Si, P, N);
X1 = [X1;ones(1, N)];
%plotData(X1, y1, m, 'Ra');
[X1p, y2] = genGaussClasses(m, Si, P, N);
X1p = [X1p;ones(1, N)];

y1(1, 101:N) = -1;
y2(1, 101:N) = -1;

w1 = [1;1;-0.5];
w2 = [1;-1;-0.5];
w3 = [-1;1;-0.5];

% Classifiers for X1
pw = perce(X1, y1, w1);
verifyVector(X1, y1, pw)
```

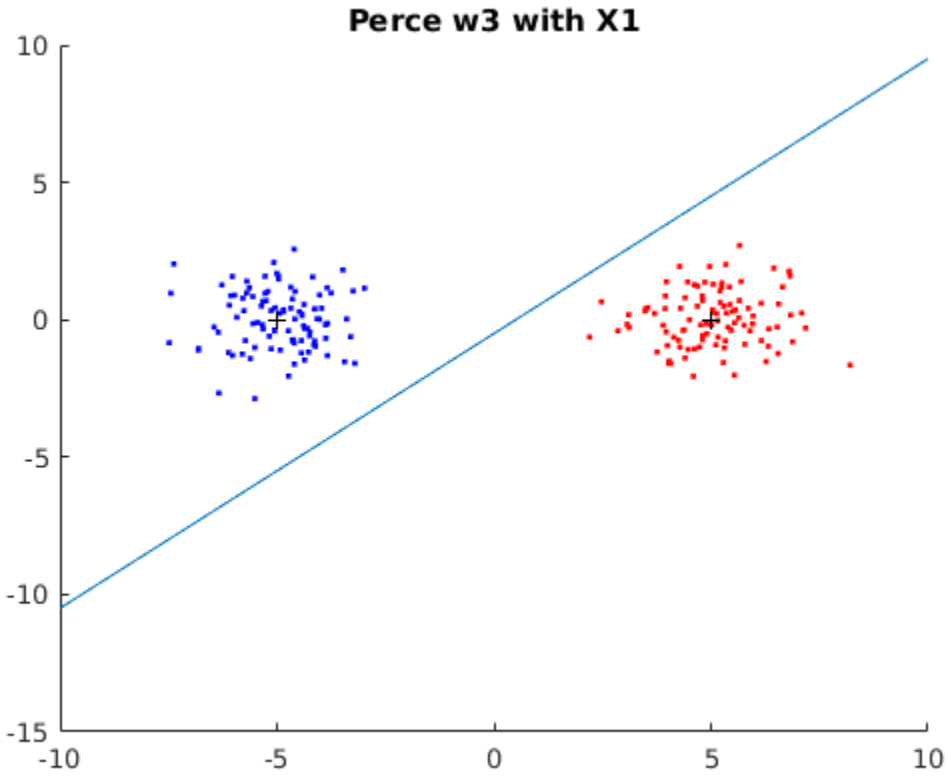
```
plotLinearClass(X1, y1, m, pw, 'Perce w1 with X1');
pw = perce(X1, y1, w2);
verifyVector(X1, y1, pw)
plotLinearClass(X1, y1, m, pw, 'Perce w2 with X1');
pw = perce(X1, y1, w3);
verifyVector(X1, y1, pw)
plotLinearClass(X1, y1, m, pw, 'Perce w3 with X1');

pw = LMSalg(X1, y1, w1);
verifyVector(X1, y1, pw)
plotLinearClass(X1, y1, m, pw, 'LMS w1 with X1');
pw = LMSalg(X1, y1, w2);
verifyVector(X1, y1, pw)
plotLinearClass(X1, y1, m, pw, 'LMS w2 with X1');
pw = LMSalg(X1, y1, w3);
verifyVector(X1, y1, pw)
plotLinearClass(X1, y1, m, pw, 'LMS w3 with X1');

pw = SSErr(X1, y1);
verifyVector(X1, y1, pw)
plotLinearClass(X1, y1, m, pw, 'SSErr with X1');

% Classifier for X1p
pw = perce(X1p, y2, w1);
verifyVector(X1p, y2, pw)
plotLinearClass(X1p, y2, m, pw, 'Perce w1 with X1p');
pw = perce(X1p, y2, w2);
verifyVector(X1p, y2, pw)
plotLinearClass(X1p, y2, m, pw, 'Perce w2 with X1p');
pw = perce(X1p, y2, w3);
verifyVector(X1p, y2, pw)
plotLinearClass(X1p, y2, m, pw, 'Perce w3 with X1p');

pw = LMSalg(X1p, y2, w1);
verifyVector(X1p, y2, pw)
plotLinearClass(X1p, y2, m, pw, 'LMS w1 with X1p');
pw = LMSalg(X1p, y2, w2);
verifyVector(X1p, y2, pw)
plotLinearClass(X1p, y2, m, pw, 'LMS w2 with X1p');
pw = LMSalg(X1p, y2, w3);
verifyVector(X1p, y2, pw)
plotLinearClass(X1p, y2, m, pw, 'LMS w3 with X1p');

pw = SSErr(X1p, y2);
verifyVector(X1p, y2, pw)
plotLinearClass(X1p, y2, m, pw, 'SSErr with X1p');


ans =

     0


ans =
```

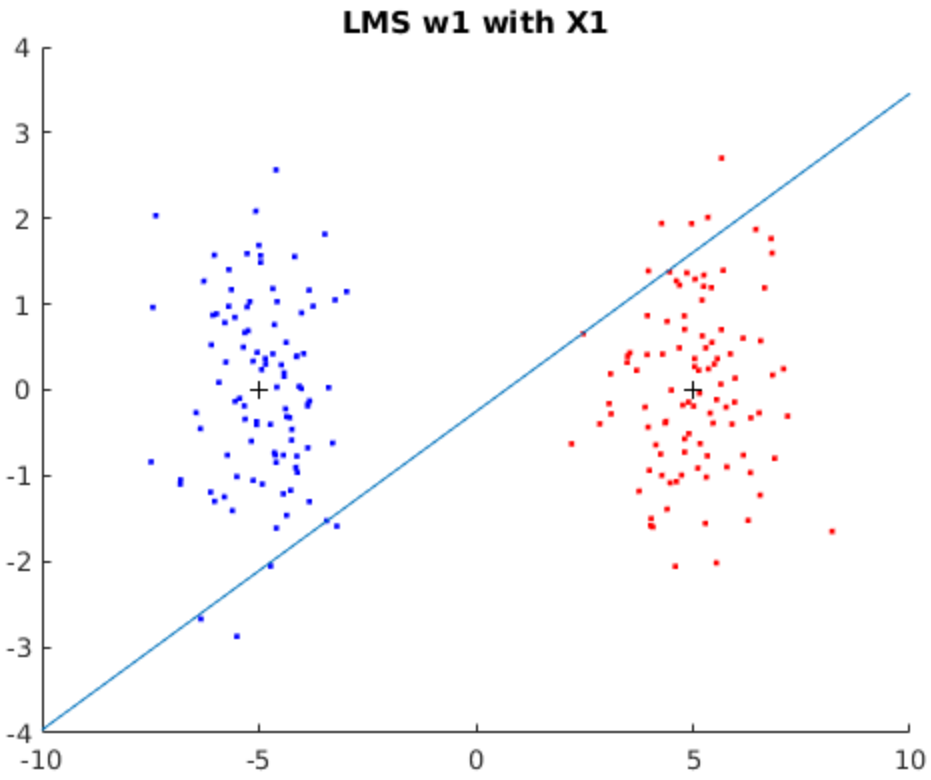
```
     0

ans =

     0

ans =

    0.0500

ans =

    0.1350

ans =

    0.2800

ans =

     0

ans =

     0

ans =

     0

ans =

     0

ans =

    0.2650

ans =

    0.0150
```

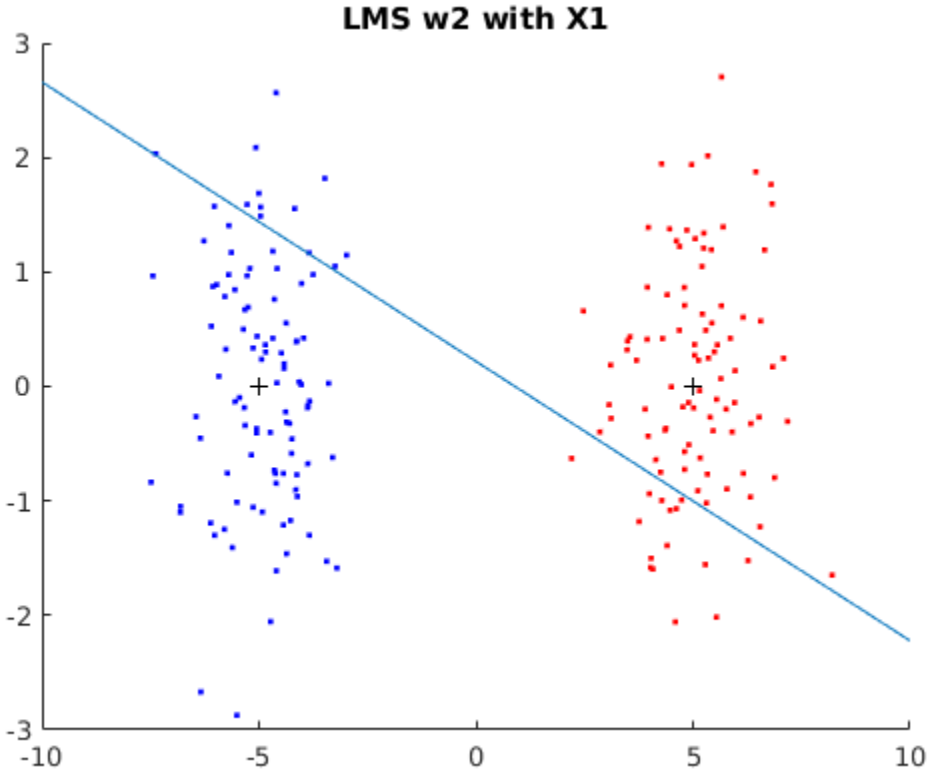
```
ans =

    0.2300

ans =

     0
```

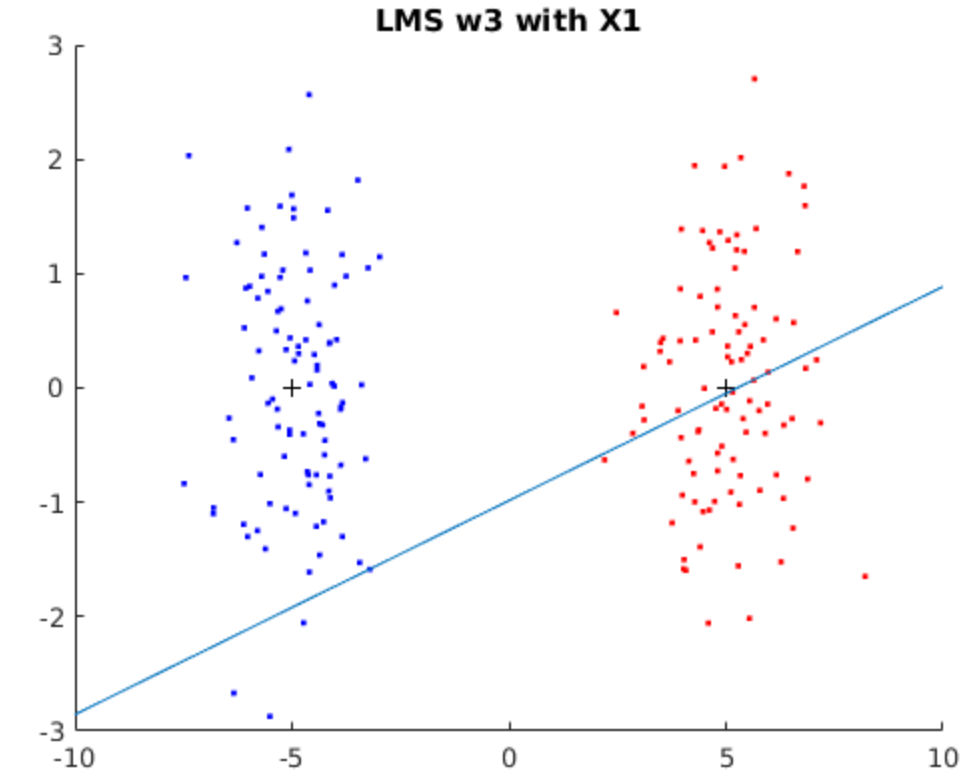


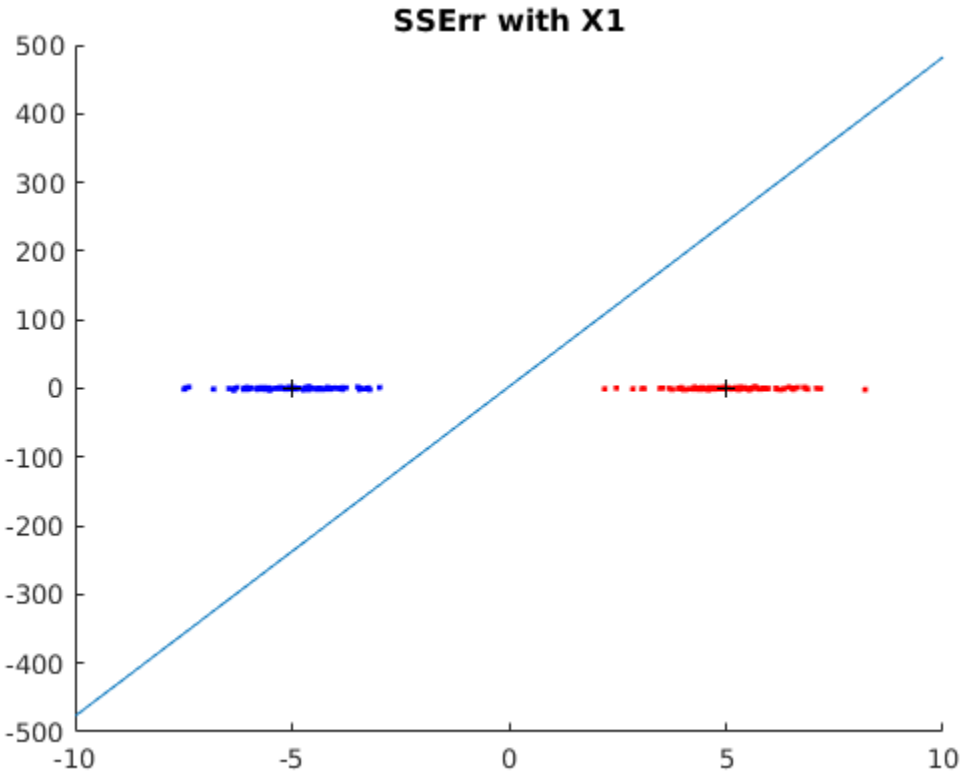

Perce w2 with X1
20
15
10
5
0
-5
-10
-15
-10
-5
0
5
10

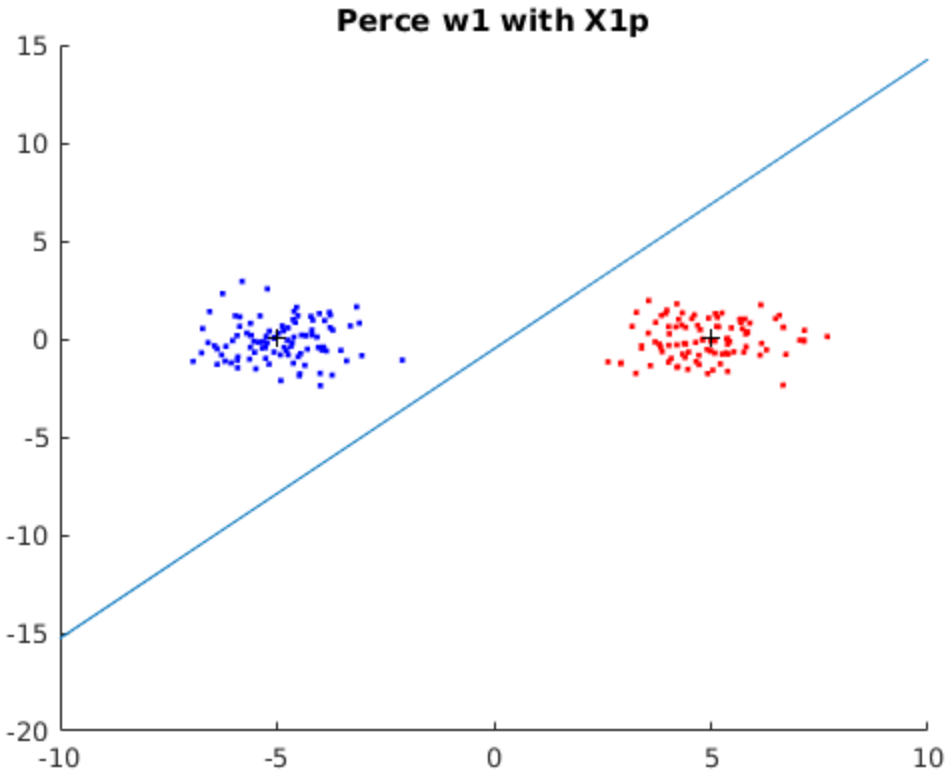

Perce w3 with X1
10
5
0
-5
-10
-15
-10
-5
0
5
10

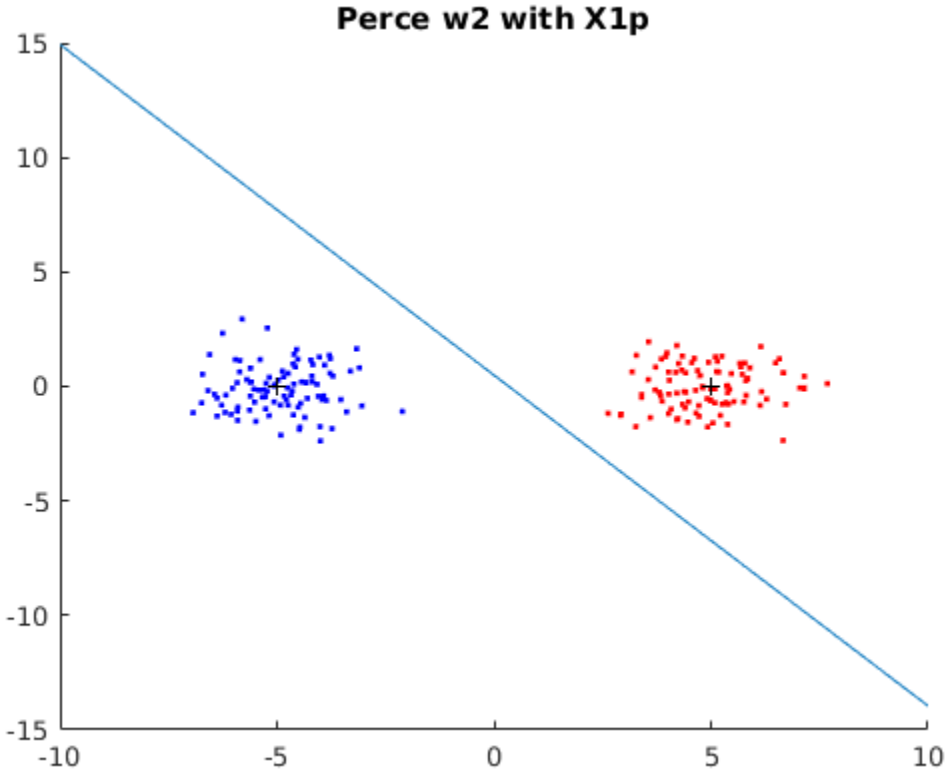

LMS w1 with X1
4
3
2
1
0
-1
-2
-3
-4
-10
-5
0
5
10

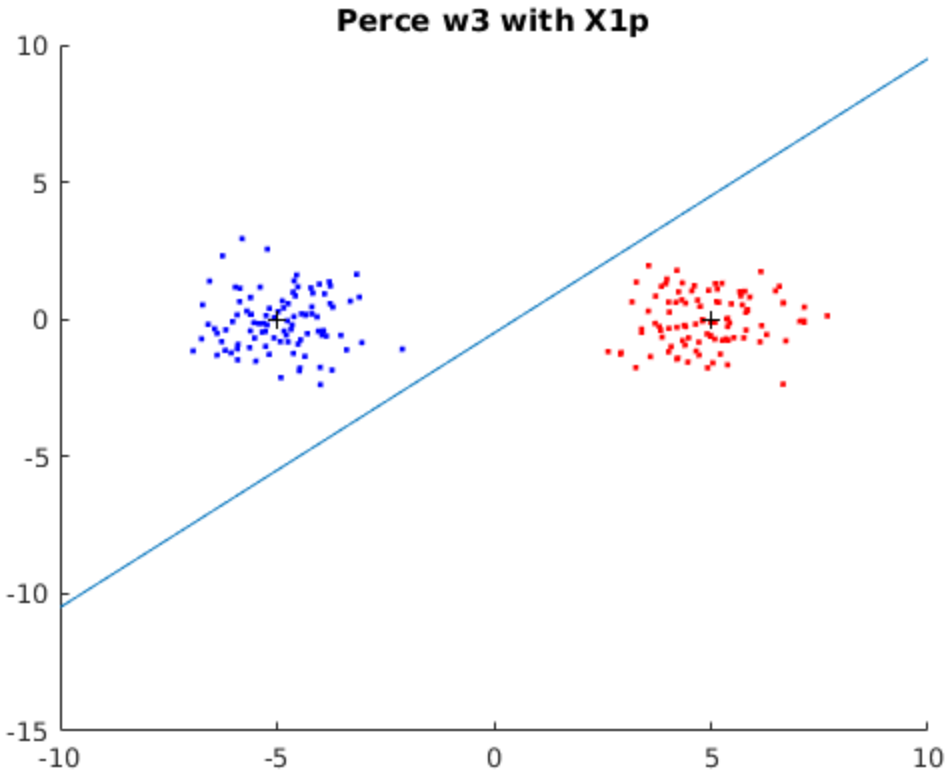

LMS w2 with X1
3
2
1
0
-1
-2
-3
-10
-5
0
5
10

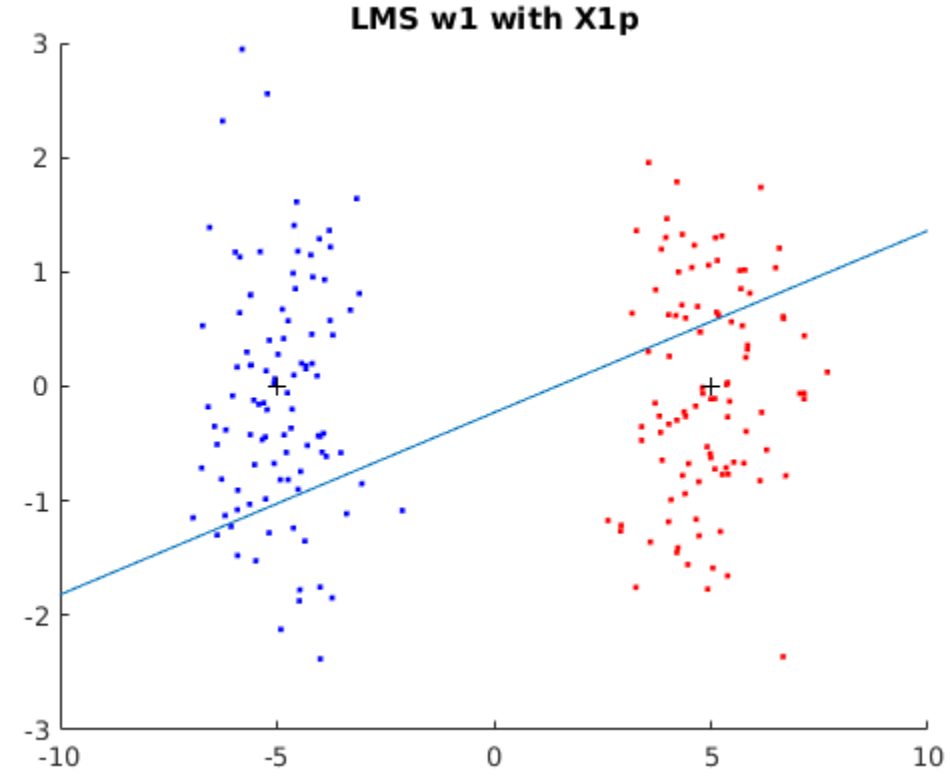

LMS w3 with X1
3
2
1
0
-1
-2
-3
-10
-5
0
5
10

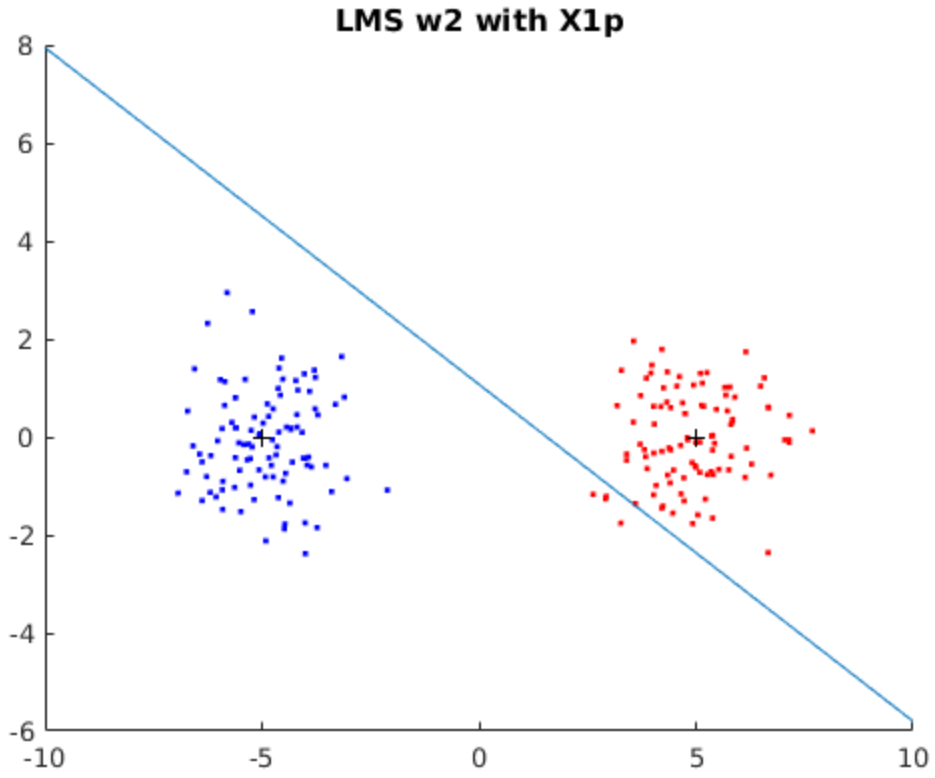

SSErr with X1
500
400
300
200
100
0
-100
-200
-300
-400
-500
-10
-5
0
5
10

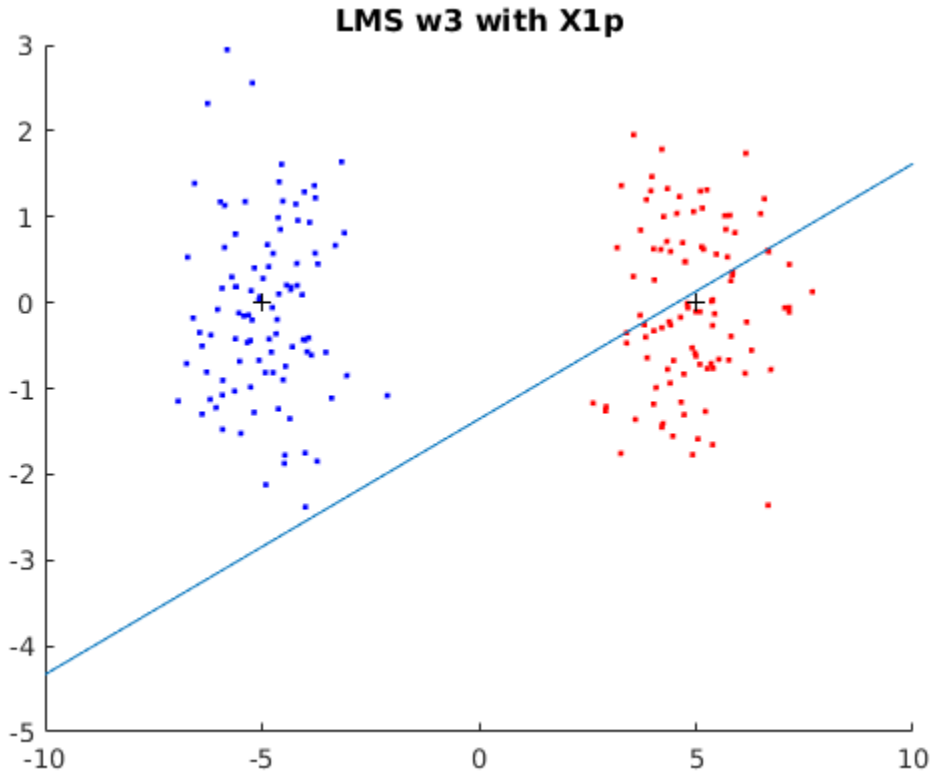

Perce w1 with X1p
15
10
5
0
-5
-10
-15
-20
-10
-5
0
5
10

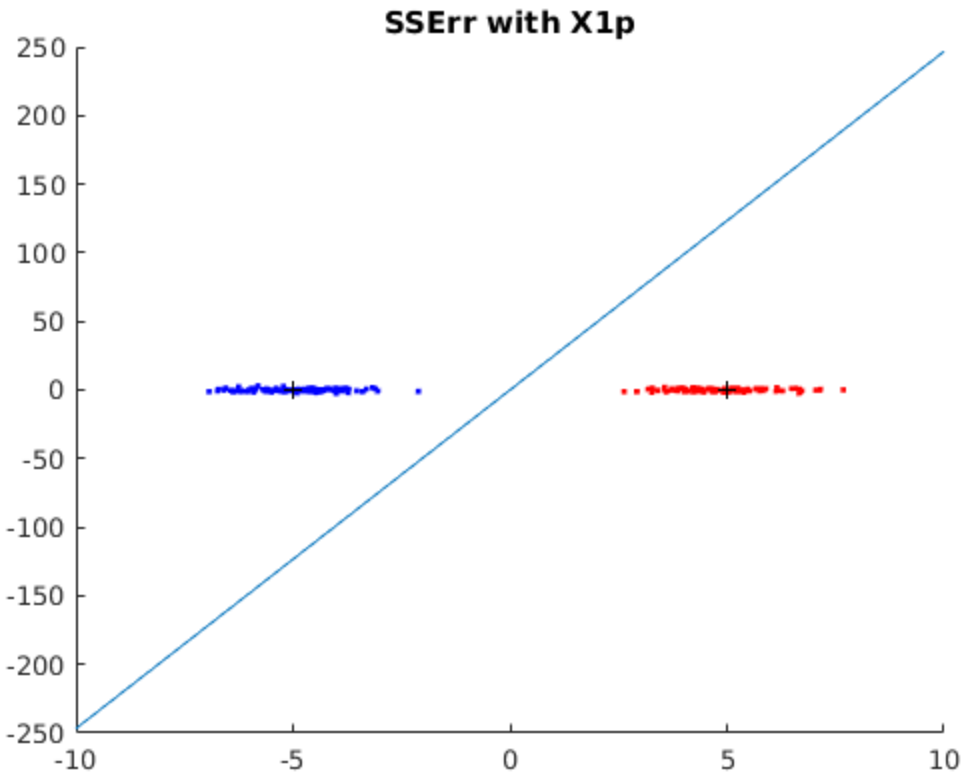

Perce w2 with X1p
15
10
5
0
-5
-10
-15
-10
-5
0
5
10


Perce w3 with X1p
10
5
0
-5
-10
-15
-10
-5
0
5
10


LMS w1 with X1p
3
2
1
0
-1
-2
-3
-10
-5
0
5
10


LMS w2 with X1p
8
6
4
2
0
-2
-4
-6
-10
-5
0
5
10


LMS w3 with X1p
3
2
1
0
-1
-2
-3
-4
-5
-10
-5
0
5
10



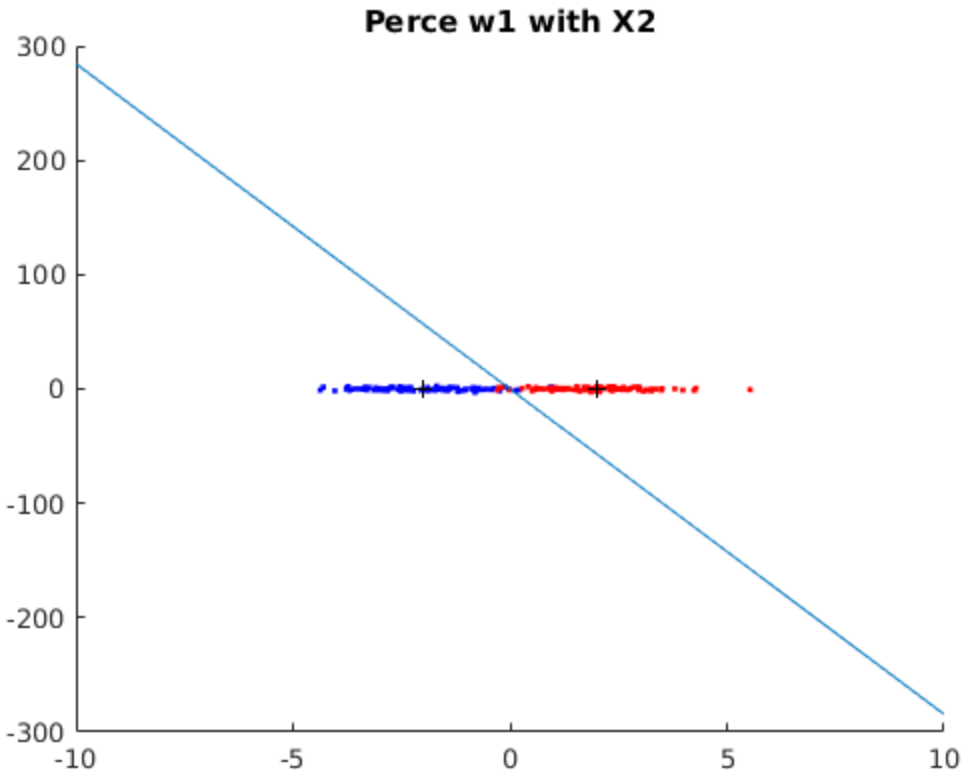
Problem 3.2

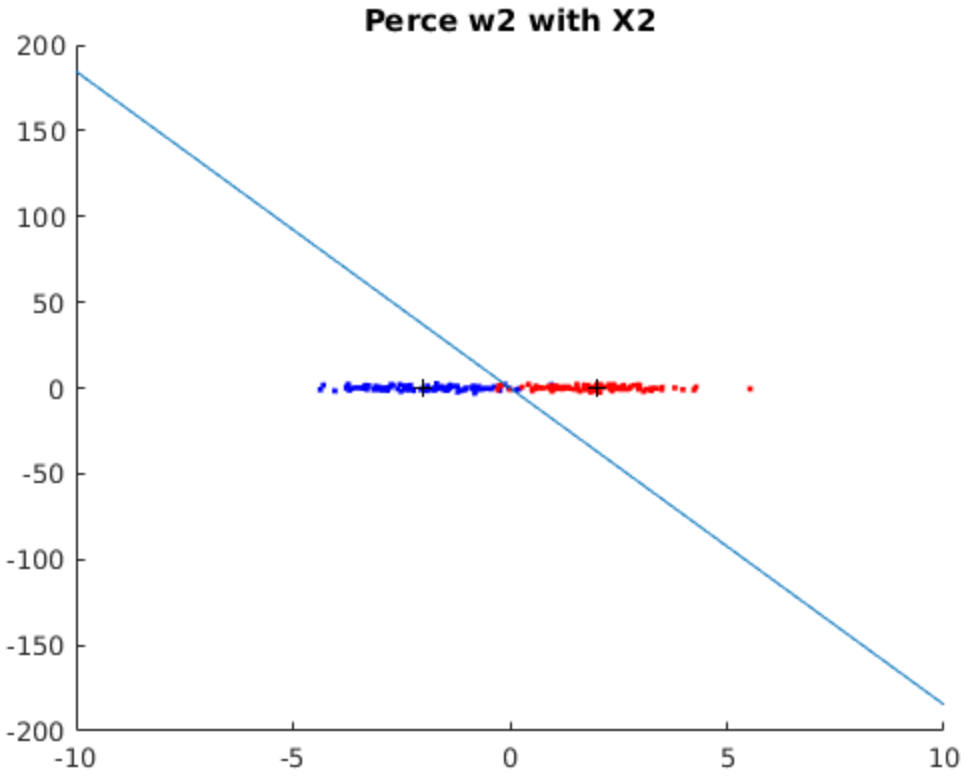
```
m = [-2 2 ;0 0];
Si = [];
P = [1/2, 1/2];
for i = 1:2
 Si(:,:,i) = [1 0; 0 1];
end


[X2, y1] = genGaussClasses(m, Si, P, N);
X2 = [X2;ones(1, N)];
[X2p, y2] = genGaussClasses(m, Si, P, N);
X2p = [X2p;ones(1, N)];

y1(1, 101:N) = -1;
y2(1, 101:N) = -1;

w1 = [1;1;-0.5];
w2 = [1;-1;-0.5];
w3 = [-1;1;-0.5];

% Classifiers for X2
pw = perce(X2, y1, w1);
verifyVector(X2, y1, pw)
```

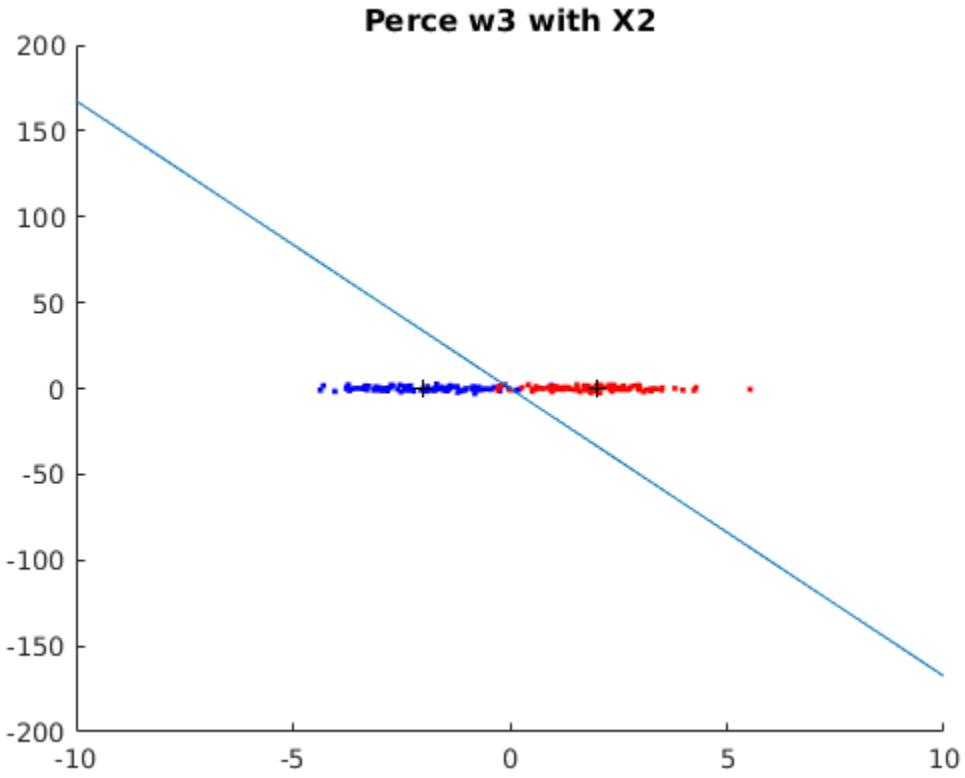
```
plotLinearClass(X2, y1, m, pw, 'Perce w1 with X2');
pw = perce(X2, y1, w2);
verifyVector(X2, y1, pw)
plotLinearClass(X2, y1, m, pw, 'Perce w2 with X2');
pw = perce(X2, y1, w3);
verifyVector(X2, y1, pw)
plotLinearClass(X2, y1, m, pw, 'Perce w3 with X2');

pw = LMSalg(X2, y1, w1);
verifyVector(X2, y1, pw)
plotLinearClass(X2, y1, m, pw, 'LMS w1 with X2');
pw = LMSalg(X2, y1, w2);
verifyVector(X2, y1, pw)
plotLinearClass(X2, y1, m, pw, 'LMS w2 with X2');
pw = LMSalg(X2, y1, w3);
verifyVector(X2, y1, pw)
plotLinearClass(X2, y1, m, pw, 'LMS w3 with X2');

pw = SSErr(X2, y1);
verifyVector(X2, y1, pw)
plotLinearClass(X2, y1, m, pw, 'SSErr with X2');

% Classifier for X2p
pw = perce(X2p, y2, w1);
verifyVector(X2p, y2, pw)
plotLinearClass(X2p, y2, m, pw, 'Perce w1 with X2p');
pw = perce(X2p, y2, w2);
verifyVector(X2p, y2, pw)
plotLinearClass(X2p, y2, m, pw, 'Perce w2 with X2p');
pw = perce(X2p, y2, w3);
verifyVector(X2p, y2, pw)
plotLinearClass(X2p, y2, m, pw, 'Perce w3 with X2p');

pw = LMSalg(X2p, y2, w1);
verifyVector(X2p, y2, pw)
plotLinearClass(X2p, y2, m, pw, 'LMS w1 with X2p');
pw = LMSalg(X2p, y2, w2);
verifyVector(X2p, y2, pw)
plotLinearClass(X2p, y2, m, pw, 'LMS w2 with X2p');
pw = LMSalg(X2p, y2, w3);
verifyVector(X2p, y2, pw)
plotLinearClass(X2p, y2, m, pw, 'LMS w3 with X2p');

pw = SSErr(X2p, y2);
verifyVector(X2p, y2, pw)
plotLinearClass(X2p, y2, m, pw, 'SSErr with X2p');


ans =

    0.0350


ans =
```

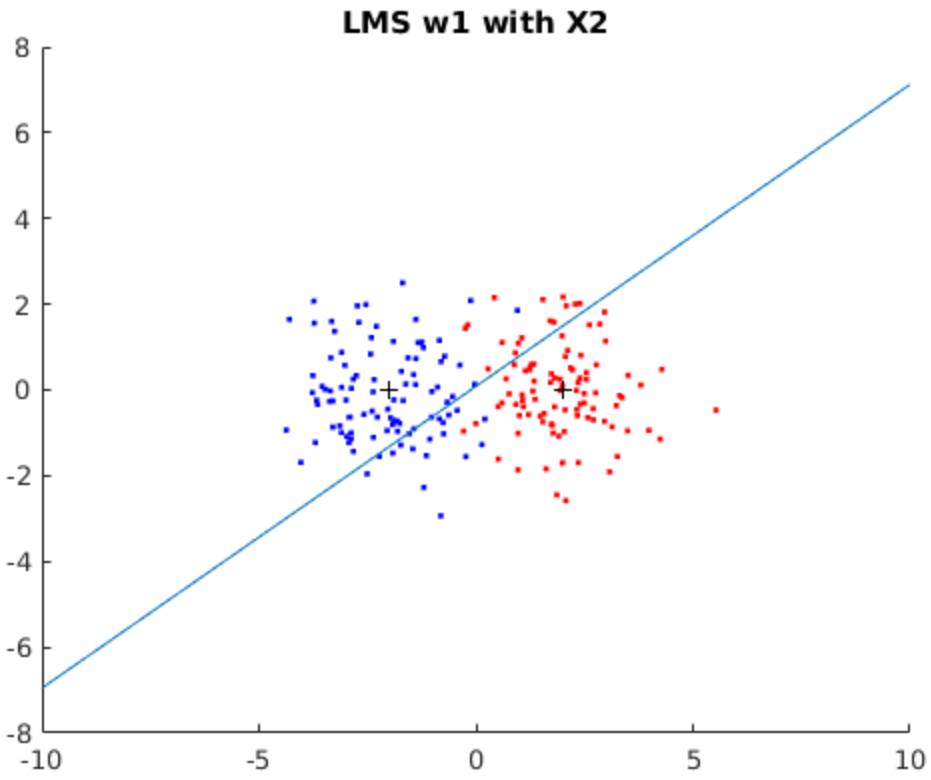
```
    0.0350

ans =

    0.0350

ans =

    0.1500

ans =

    0.1250

ans =

    0.1350

ans =

    0.0400

ans =

    0.0100

ans =

    0.0100

ans =

    0.0150

ans =

    0.0300

ans =

    0.2450
```

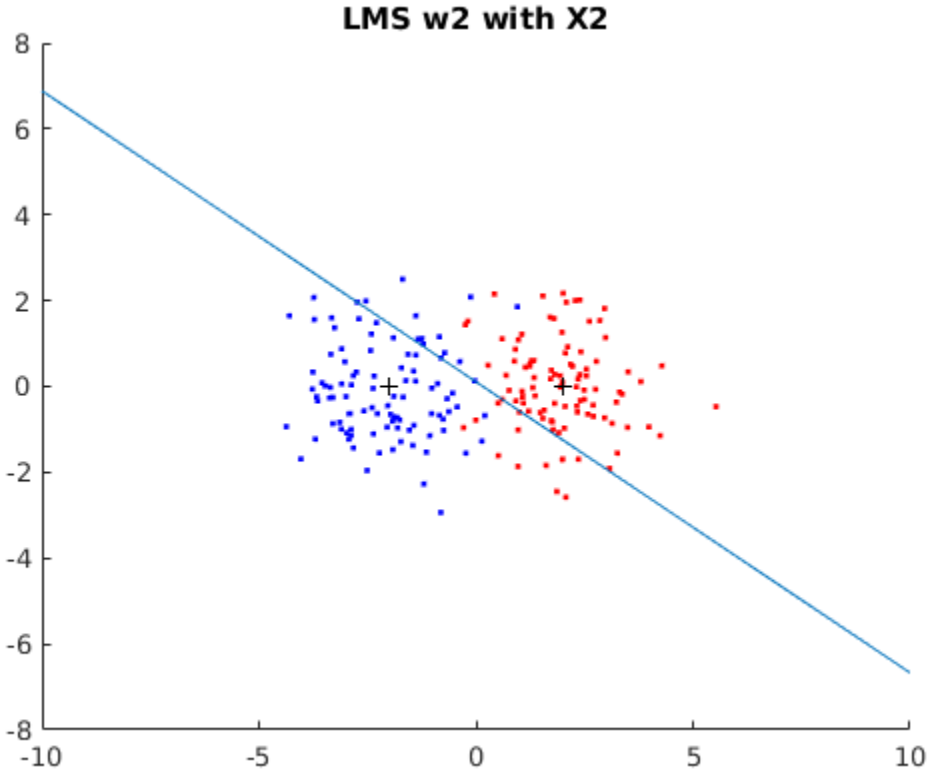
```
ans =

    0.1650

ans =

    0.0150
```

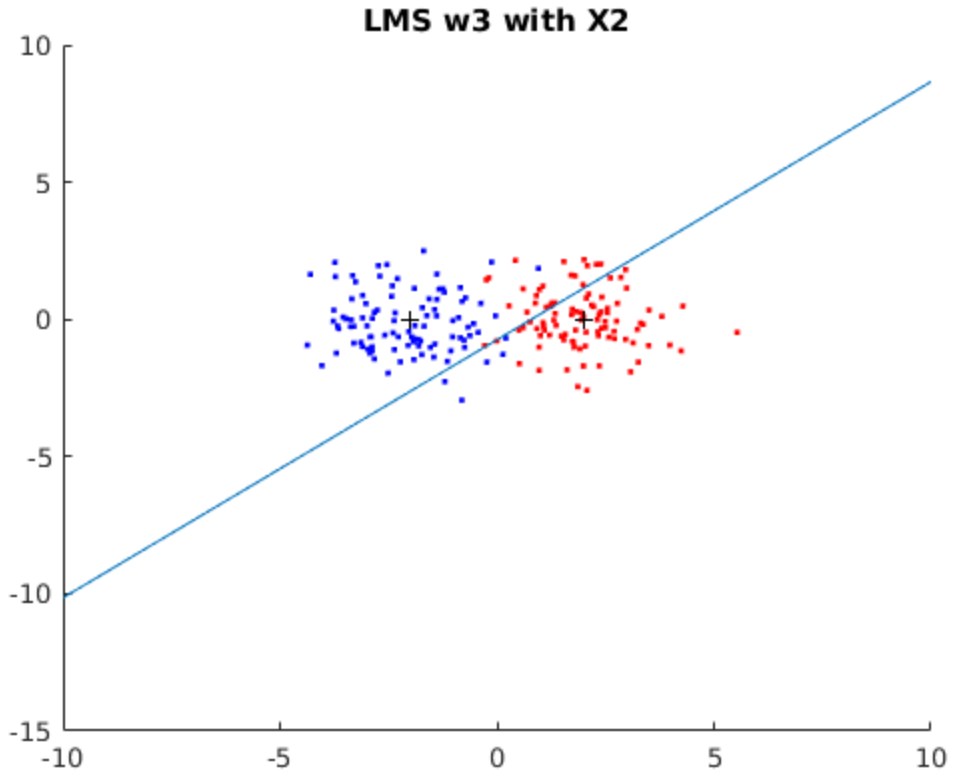


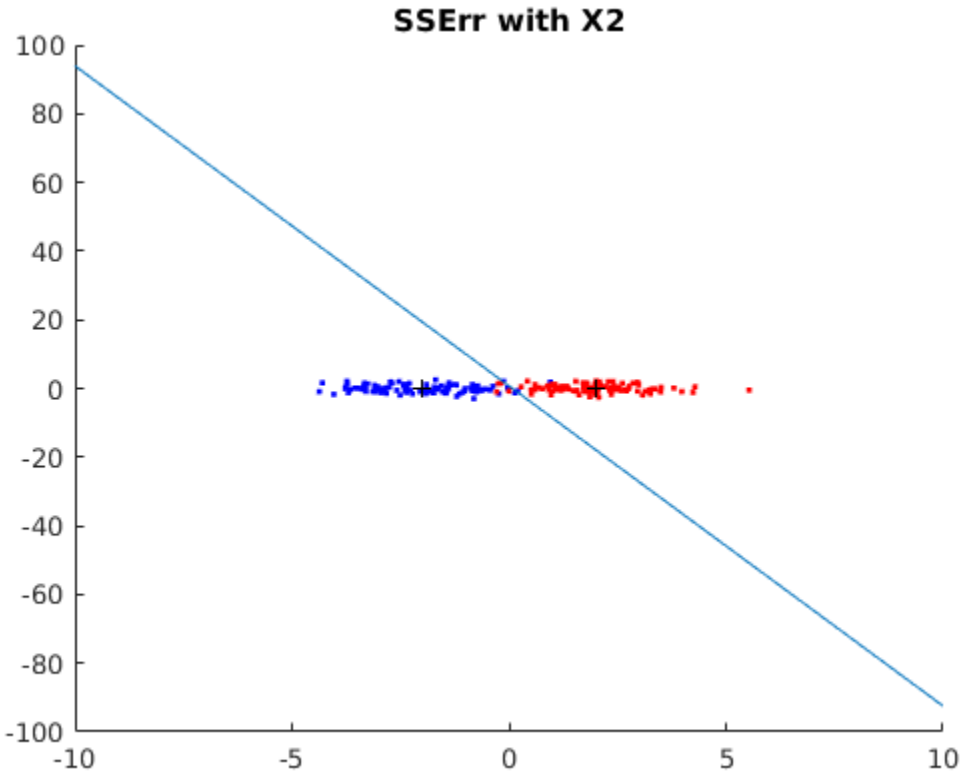

Perce w2 with X2
200
150
100
50
0
-50
-100
-150
-200
-10
-5
0
5
10

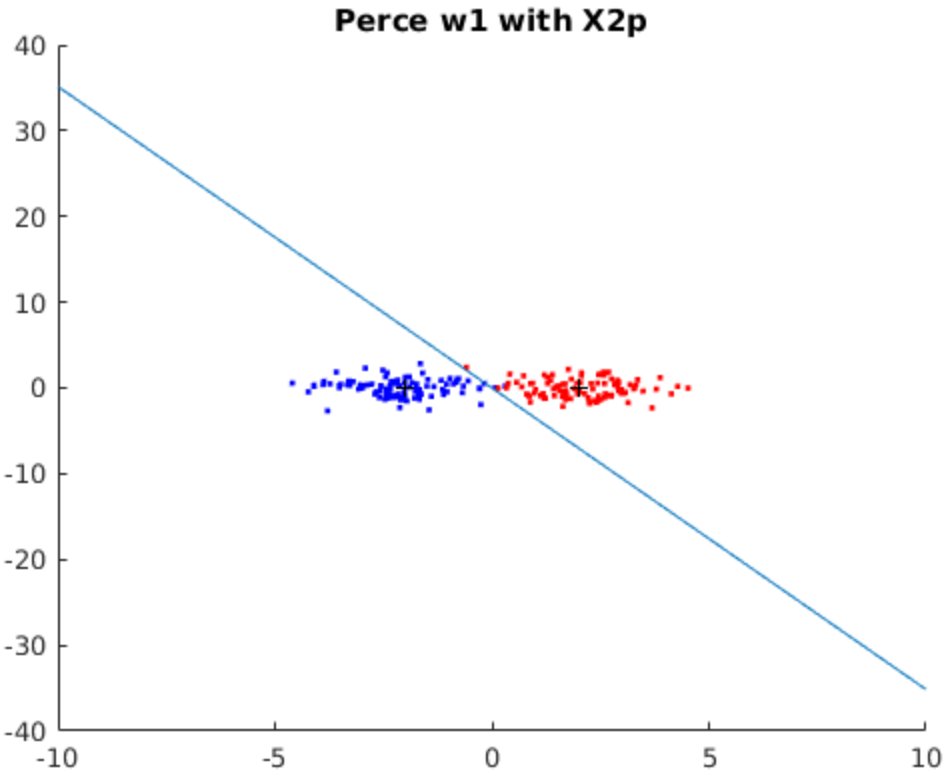

Perce w3 with X2
200
150
100
50
0
-50
-100
-150
-200
-10
-5
0
5
10

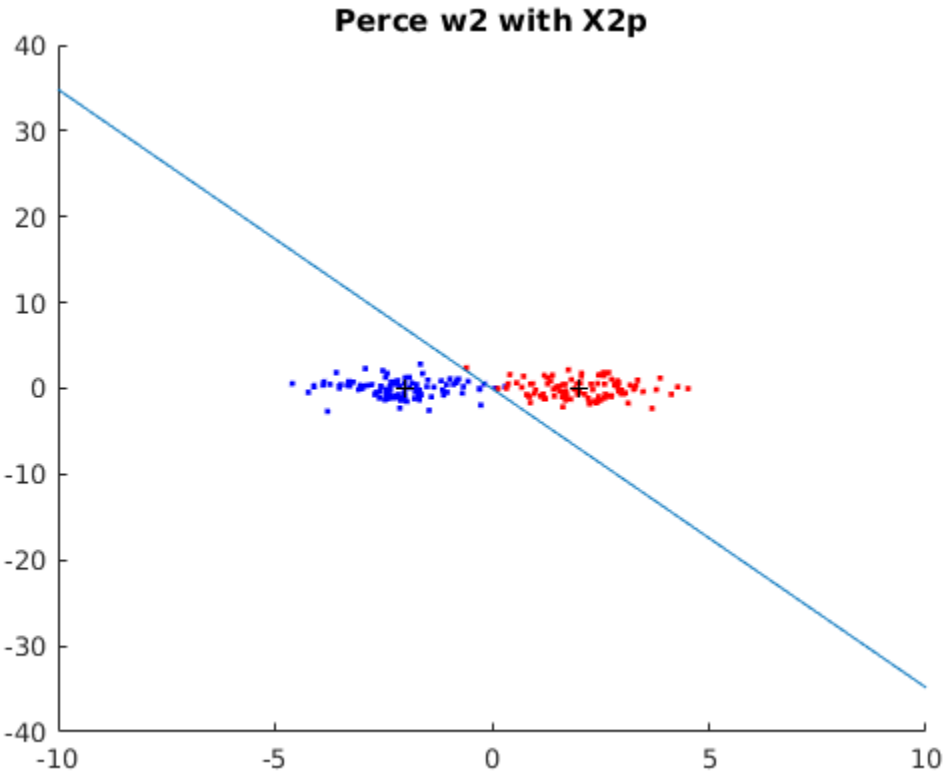

LMS w1 with X2
8
6
4
2
0
-2
-4
-6
-8
-10
-5
0
5
10

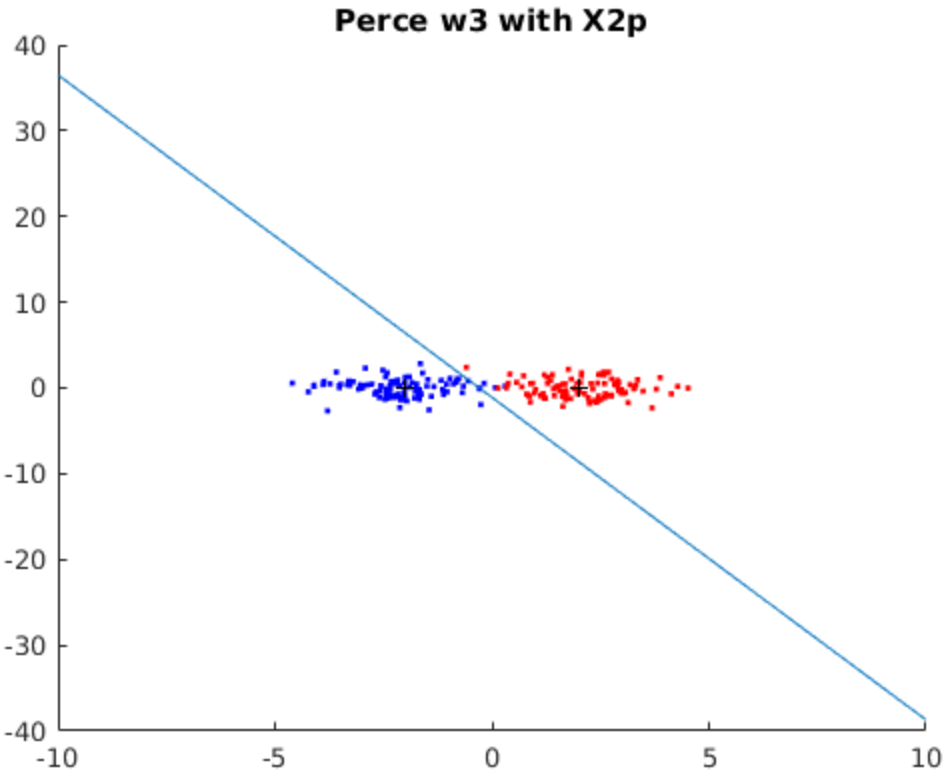

LMS w2 with X2
8
6
4
2
0
-2
-4
-6
-8
-10
-5
0
5
10

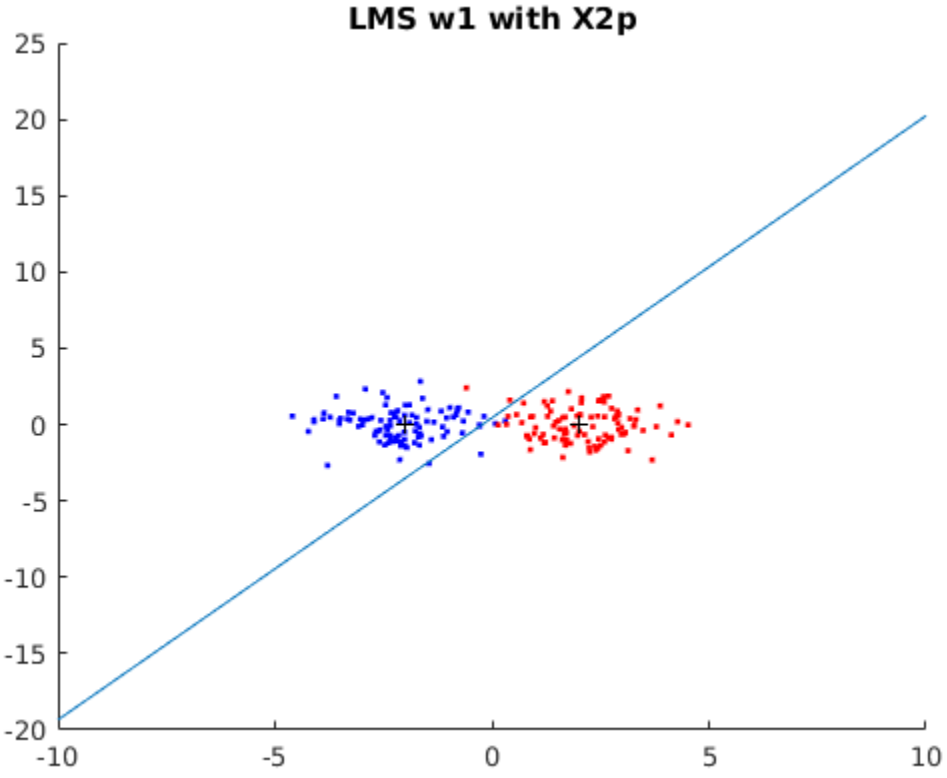

LMS w3 with X2
10
5
0
-5
-10
-15
-10
-5
0
5
10

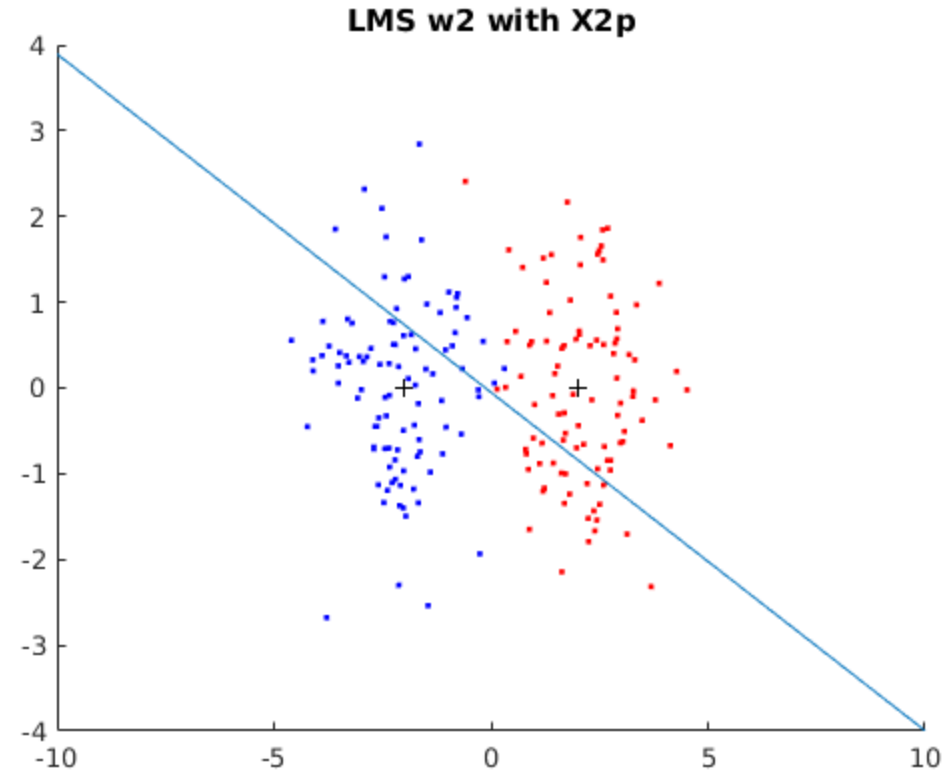

SSErr with X2
100
80
60
40
20
0
-20
-40
-60
-80
-100
-10
-5
0
5
10

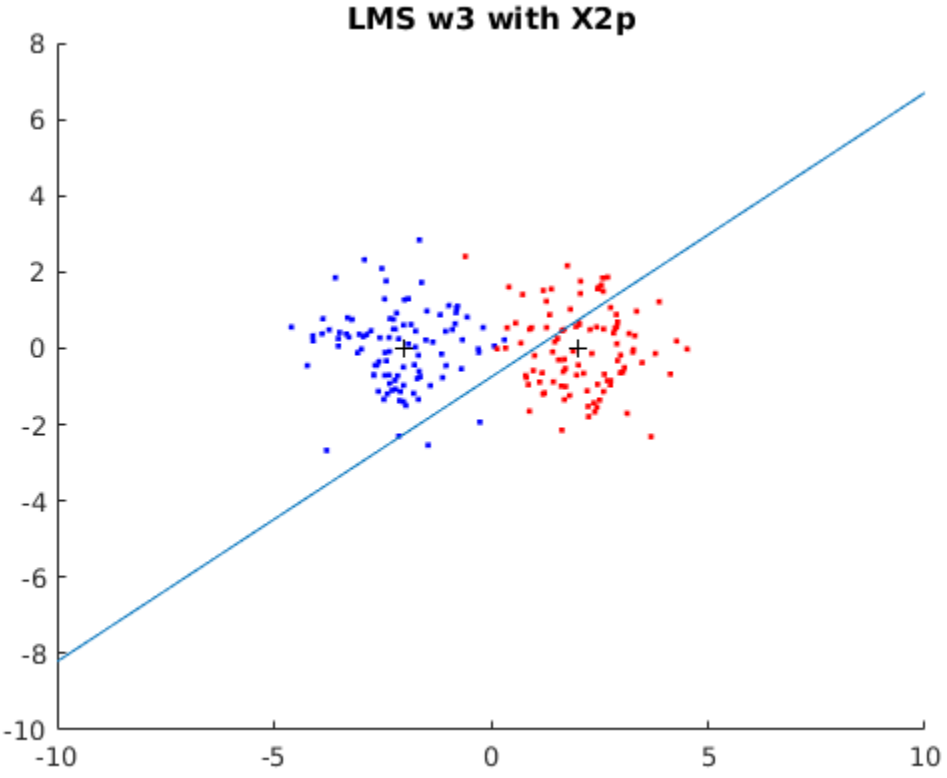

Perce w1 with X2p
40
30
20
10
0
-10
-20
-30
-40
-10
-5
0
5
10

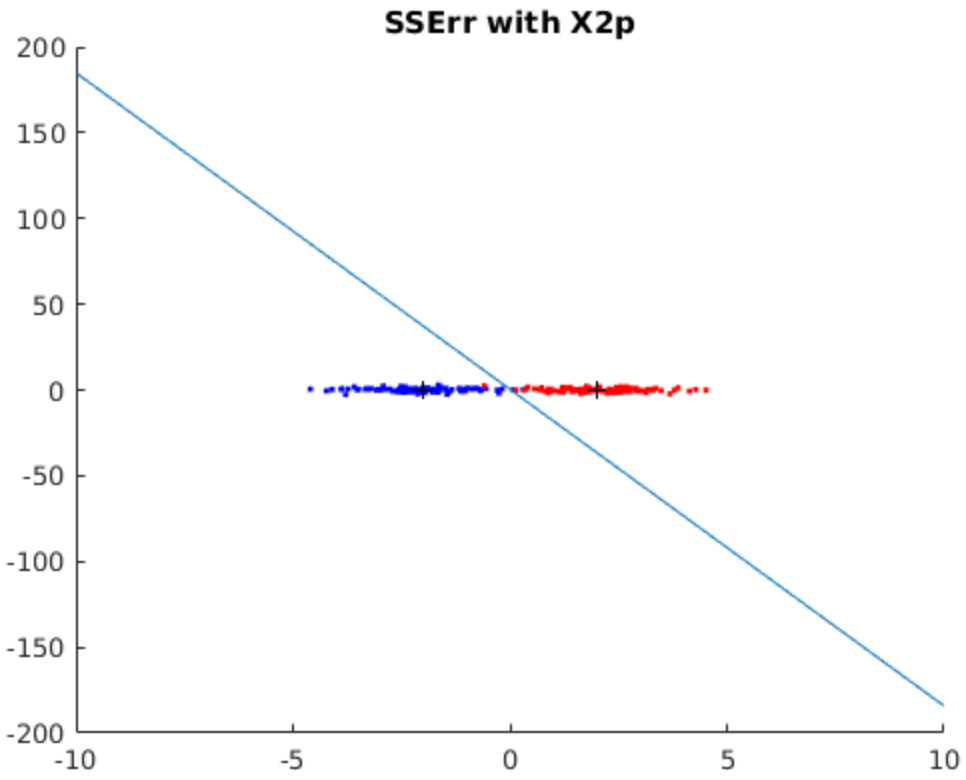

Perce w2 with X2p
40
30
20
10
0
-10
-20
-30
-40
-10
-5
0
5
10


Perce w3 with X2p
40
30
20
10
0
-10
-20
-30
-40
-10
-5
0
5
10


LMS w1 with X2p
25
20
15
10
5
0
-5
-10
-15
-20
-10
-5
0
5
10


LMS w2 with X2p
4
3
2
1
0
-1
-2
-3
-4
-10
-5
0
5
10


LMS w3 with X2p
8
6
4
2
0
-2
-4
-6
-8
-10
-10
-5
0
5
10



Published with MATLAB® R2017a