

---

## Table of Contents

.....	1
fetching data .....	1
feasible start point for optimization .....	1
plotting training data for 2d case .....	2
plotting separating hyperplane .....	3

```
clear all; clc;
% t = 10; % log barrier function scaling parameter
```

## fetching data

```
[xt, yt] = getData();
siz = size(xt);
numData = siz(1);
dimData = siz(2);
```

## feasible start point for optimization

```
x0 = get_start_point();
disp("-----Feasible start point optimization result (w; b;
    s_min)-----");
disp(x0);
```

```
x0=x0(1:dimData+1);
% x0=[-1.3574;5.1205;1.8035];
% x0=[-1.0024;5.1205;1.8035];
disp("-----Feasible start point (w; b)-----");
disp(x0);
```

```
options = optimoptions(@fminunc,'Algorithm','quasi-newton');
[x,fval,exitflag,output] = fminunc(@objective,x0,options);
x
```

*Local minimum found that satisfies the constraints.*

*Optimization completed because the objective function is non-decreasing in  
feasible directions, to within the default value of the optimality  
tolerance,  
and constraints are satisfied to within the default value of the  
constraint tolerance.*

---

```
-----Feasible start point optimization result (w; b;  
s_min)-----  
-11.7019  
58.7848  
7.0786  
-2.0000
```

```
-----Feasible start point (w; b)-----  
-11.7019  
58.7848  
7.0786
```

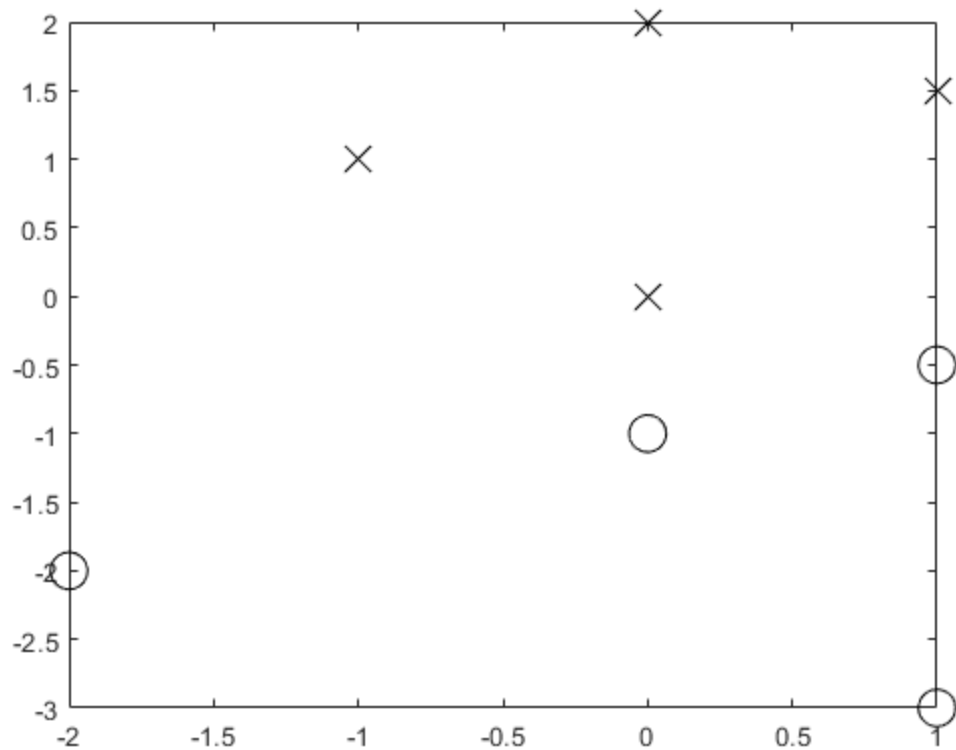
*Local minimum found.*

*Optimization completed because the size of the gradient is less than the default value of the optimality tolerance.*

```
x =  
  
-1.3574  
5.1206  
1.8035
```

## plotting training data for 2d case

```
fig = figure;  
hold off;  
msize=14;  
for i = 1:numData  
    if(yt(i)==1)  
        plot(xt(i,1),xt(i,2), 'xblack', 'MarkerSize',msize);  
    else  
        plot(xt(i,1),xt(i,2), 'oblack', 'MarkerSize',msize);  
    end  
    hold on;  
end
```

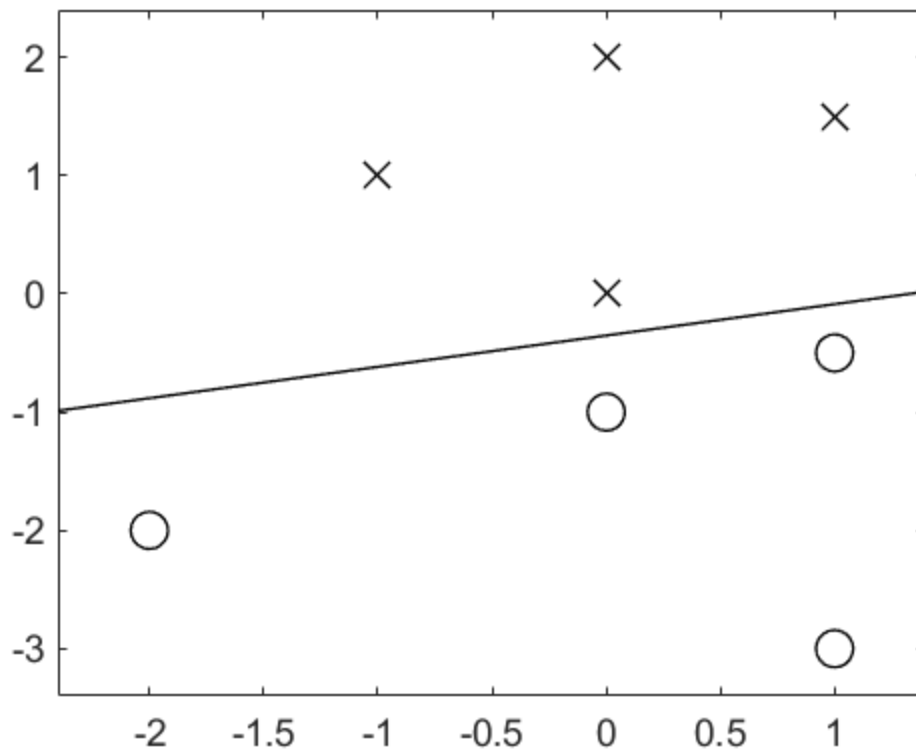


## plotting separating hyperplane

```
margin = 0.4;
xMin = min(xt(:,1))-margin;
xMax = max(xt(:,1))+margin;
yMin = -(xMin*x(1)+x(3))/x(2);
yMax = -(xMax*x(1)+x(3))/x(2);
plot([xMin, xMax],[yMin, yMax], 'black');

set(gca, 'FontSize', 14)
set(findall(gca, 'Type', 'Line'), 'LineWidth', 1);
xlim([xMin, xMax]);
ylim([min(xt(:,2))-margin, max(xt(:,2))+margin]));

print(fig, 'images/
separating_hyperplane_linearly_separable_data_log_barrier_method_t_1', '-
dpng');
```



*Published with MATLAB® R2017b*