



Unit 1 - Sem 4 - 22MAT230

Mathematics for Computing

Dr Sunil Kumar S and Prof K P Soman

School of Artificial Intelligence

Amrita Vishwa Vidyapeetham

If you find any mistakes or have any comments to share,

I would be grateful to receive them at s_sunilkumar@cb.amrita.edu

ℓ_1 norm optimized solution of the system of equations $A\mathbf{x} = \mathbf{b}$ is sparse.

Compare least ℓ_1 and ℓ_2 norm solution of $A\mathbf{x} = \mathbf{b}$

```
clearvars
```

Construct A and \mathbf{b}

```
n = 10;  
m = 3;  
A = randi([-5,5],m,n);  
b = A*randi([-3,3],n,1)
```

```
b = 3x1  
    19  
   -39  
   -11
```

Find the least ℓ_2 norm solution (using pseudo inverse)

```
x_2 = pinv(A)*b;
```

Find the least ℓ_1 norm solution (using ADMM)

```
maxIter = 400;  
rho = 1.6;
```

```

B1 = A'*pinv(A*A');
B1b = B1*b;
B2 = B1*A;
converged = false;
dz = 1e-6;

```

School of AI, AVV

Initialize the **z** and **u** vectors

```

Z0 = rand(n,1);
U0 = rand(n,1);

for i = 1:maxIter
    % X update - using Shrinkage
    c = Z0 - U0;
    X1 = c - sign(c)/rho;
    X1(sign(X1) ~= sign(c)) = 0;
    % Z update - using Projection
    mu = X1 + U0;
    Z1 = (eye(n) - B2)*mu + B1b;
    % U update - using gradient
    U1 = U0 + (X1 - Z1);
    if(norm(Z1-Z0) <= dz)
        converged = true;
        break
    end
    fprintf("iter : %d \t dz : %f dxz : %f \n",i,norm(Z1-Z0),norm(X1-Z0));
    % fprintf("iter : %d \t dz : %f\n",i,norm(X1-Z0));
    Z0 = Z1;
    U0 = U1;
end

```

```

iter : 1      dz : 3.258914 dxz : 1.766320
iter : 2      dz : 0.836954 dxz : 1.782286
iter : 3      dz : 0.479248 dxz : 0.561535
iter : 4      dz : 0.414353 dxz : 0.432155
iter : 5      dz : 0.390348 dxz : 0.397540
iter : 6      dz : 0.341284 dxz : 0.342971
iter : 7      dz : 0.183572 dxz : 0.229237
iter : 8      dz : 0.129540 dxz : 0.184025
iter : 9      dz : 0.142058 dxz : 0.159554
iter : 10     dz : 0.142394 dxz : 0.146112
iter : 11     dz : 0.135763 dxz : 0.138736
iter : 12     dz : 0.131979 dxz : 0.134605
iter : 13     dz : 0.129524 dxz : 0.132208
iter : 14     dz : 0.128253 dxz : 0.130756
iter : 15     dz : 0.128520 dxz : 0.129840
iter : 16     dz : 0.129006 dxz : 0.129242
iter : 17     dz : 0.128812 dxz : 0.128841
iter : 18     dz : 0.128274 dxz : 0.128567
iter : 19     dz : 0.127959 dxz : 0.128378
iter : 20     dz : 0.127965 dxz : 0.128246
iter : 21     dz : 0.124088 dxz : 0.124234
iter : 22     dz : 0.094803 dxz : 0.099062
iter : 23     dz : 0.092867 dxz : 0.095436
iter : 24     dz : 0.092873 dxz : 0.094293
iter : 25     dz : 0.093237 dxz : 0.093720
iter : 26     dz : 0.093289 dxz : 0.093364
iter : 27     dz : 0.093034 dxz : 0.093126
iter : 28     dz : 0.092710 dxz : 0.092960
iter : 29     dz : 0.092547 dxz : 0.092843

```

| | | |
|-----------|---------------|----------------|
| iter : 30 | dz : 0.076666 | dxz : 0.078802 |
| iter : 31 | dz : 0.057250 | dxz : 0.068590 |
| iter : 32 | dz : 0.046646 | dxz : 0.062415 |
| iter : 33 | dz : 0.045959 | dxz : 0.057906 |
| iter : 34 | dz : 0.048967 | dxz : 0.054518 |
| iter : 35 | dz : 0.050211 | dxz : 0.051948 |
| iter : 36 | dz : 0.049069 | dxz : 0.049994 |
| iter : 37 | dz : 0.046914 | dxz : 0.048510 |
| iter : 38 | dz : 0.045079 | dxz : 0.047385 |
| iter : 39 | dz : 0.044227 | dxz : 0.046530 |
| iter : 40 | dz : 0.044280 | dxz : 0.045882 |
| iter : 41 | dz : 0.027457 | dxz : 0.033487 |
| iter : 42 | dz : 0.015614 | dxz : 0.028605 |
| iter : 43 | dz : 0.008678 | dxz : 0.024729 |
| iter : 44 | dz : 0.011173 | dxz : 0.021526 |
| iter : 45 | dz : 0.015022 | dxz : 0.018810 |
| iter : 46 | dz : 0.015611 | dxz : 0.016472 |
| iter : 47 | dz : 0.012829 | dxz : 0.014442 |
| iter : 48 | dz : 0.008041 | dxz : 0.012671 |
| iter : 49 | dz : 0.003531 | dxz : 0.011120 |
| iter : 50 | dz : 0.004002 | dxz : 0.009762 |
| iter : 51 | dz : 0.006337 | dxz : 0.008571 |
| iter : 52 | dz : 0.007091 | dxz : 0.007525 |
| iter : 53 | dz : 0.006172 | dxz : 0.006608 |
| iter : 54 | dz : 0.004142 | dxz : 0.005803 |
| iter : 55 | dz : 0.001852 | dxz : 0.005096 |
| iter : 56 | dz : 0.001354 | dxz : 0.004475 |
| iter : 57 | dz : 0.002578 | dxz : 0.003930 |
| iter : 58 | dz : 0.003161 | dxz : 0.003452 |
| iter : 59 | dz : 0.002936 | dxz : 0.003032 |
| iter : 60 | dz : 0.002118 | dxz : 0.002663 |
| iter : 61 | dz : 0.001046 | dxz : 0.002339 |
| iter : 62 | dz : 0.000412 | dxz : 0.002055 |
| iter : 63 | dz : 0.000995 | dxz : 0.001805 |
| iter : 64 | dz : 0.001375 | dxz : 0.001585 |
| iter : 65 | dz : 0.001373 | dxz : 0.001393 |
| iter : 66 | dz : 0.001066 | dxz : 0.001224 |
| iter : 67 | dz : 0.000593 | dxz : 0.001075 |
| iter : 68 | dz : 0.000153 | dxz : 0.000945 |
| iter : 69 | dz : 0.000354 | dxz : 0.000830 |
| iter : 70 | dz : 0.000579 | dxz : 0.000729 |
| iter : 71 | dz : 0.000629 | dxz : 0.000641 |
| iter : 72 | dz : 0.000526 | dxz : 0.000563 |
| iter : 73 | dz : 0.000328 | dxz : 0.000495 |
| iter : 74 | dz : 0.000105 | dxz : 0.000435 |
| iter : 75 | dz : 0.000108 | dxz : 0.000382 |
| iter : 76 | dz : 0.000234 | dxz : 0.000336 |
| iter : 77 | dz : 0.000281 | dxz : 0.000295 |
| iter : 78 | dz : 0.000254 | dxz : 0.000259 |
| iter : 79 | dz : 0.000175 | dxz : 0.000228 |
| iter : 80 | dz : 0.000075 | dxz : 0.000200 |
| iter : 81 | dz : 0.000022 | dxz : 0.000176 |
| iter : 82 | dz : 0.000089 | dxz : 0.000155 |
| iter : 83 | dz : 0.000122 | dxz : 0.000136 |
| iter : 84 | dz : 0.000120 | dxz : 0.000120 |
| iter : 85 | dz : 0.000090 | dxz : 0.000105 |
| iter : 86 | dz : 0.000047 | dxz : 0.000093 |
| iter : 87 | dz : 0.000003 | dxz : 0.000081 |
| iter : 88 | dz : 0.000031 | dxz : 0.000072 |
| iter : 89 | dz : 0.000051 | dxz : 0.000063 |
| iter : 90 | dz : 0.000055 | dxz : 0.000055 |
| iter : 91 | dz : 0.000045 | dxz : 0.000049 |
| iter : 92 | dz : 0.000028 | dxz : 0.000043 |
| iter : 93 | dz : 0.000008 | dxz : 0.000038 |
| iter : 94 | dz : 0.000010 | dxz : 0.000033 |
| iter : 95 | dz : 0.000021 | dxz : 0.000029 |
| iter : 96 | dz : 0.000024 | dxz : 0.000026 |
| iter : 97 | dz : 0.000022 | dxz : 0.000022 |

```

iter : 98      dz : 0.000015 dxz : 0.000020
iter : 99      dz : 0.000007 dxz : 0.000017
iter : 100     dz : 0.000002 dxz : 0.000015
iter : 101     dz : 0.000008 dxz : 0.000013
iter : 102     dz : 0.000011 dxz : 0.000012
iter : 103     dz : 0.000010 dxz : 0.000010
iter : 104     dz : 0.000008 dxz : 0.000009
iter : 105     dz : 0.000004 dxz : 0.000008
iter : 106     dz : 0.000001 dxz : 0.000007
iter : 107     dz : 0.000003 dxz : 0.000006
iter : 108     dz : 0.000004 dxz : 0.000005
iter : 109     dz : 0.000005 dxz : 0.000005
iter : 110     dz : 0.000004 dxz : 0.000004
iter : 111     dz : 0.000003 dxz : 0.000004
iter : 112     dz : 0.000001 dxz : 0.000003

```

```

x_1 = X1;

if(converged == true)
    fprintf("L1 norm optimized solution found after %d iterations. \n",i);
elseif(converged == false)
    fprintf("Convergence failed\n")
    fprintf("Try : increasing the no: iteration, increasing the
convergence tol value, changing rho\n");
end

```

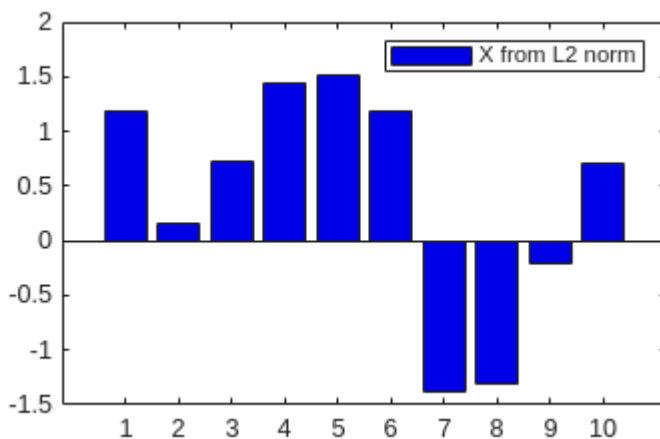
L1 norm optimized solution found after 113 iterations.

Plot to check the solution vector X and surrogate variable Z

```

b2 = bar(x_2);%hold on;
b2.FaceColor = [0 0 0.9];
legend("X from L2 norm")

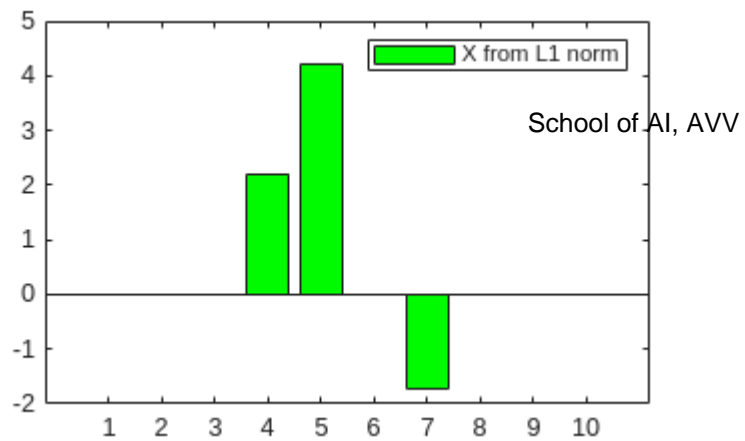
```



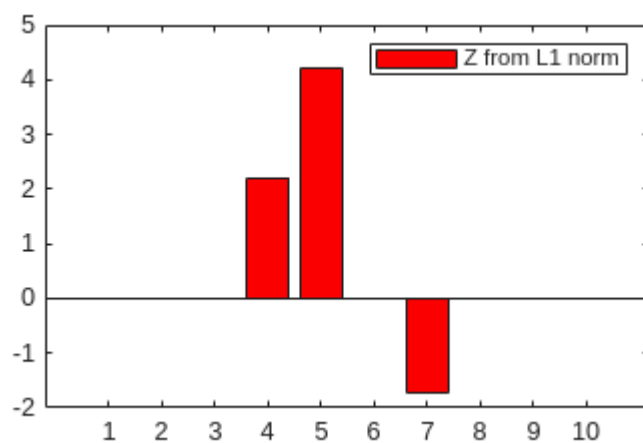
```

b1 = bar(x_1);%hold off
b1.FaceColor = [0 0.98 0];
legend("X from L1 norm")

```



```
b1 = bar(Z1);%hold off
b1.FaceColor = [0.98 0 0];
legend("Z from L1 norm")
```



```
plot(X1,'k.');
```

```
hold on
```

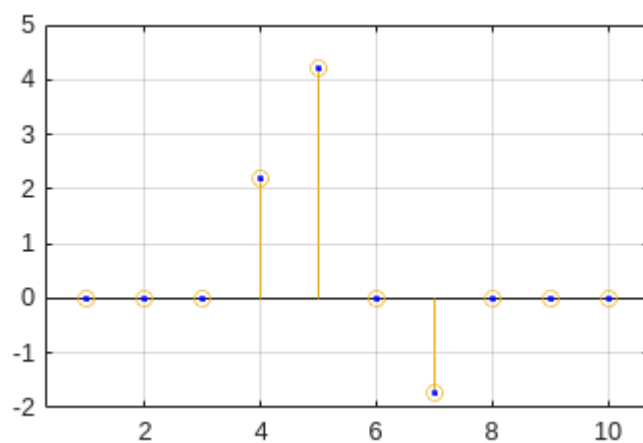
```
stem(X1)
```

```
stem(Z1)
```

```
plot(Z1,'b.');
```

```
hold off
```

```
grid on
```



Check the error vector $Ax - b$

```
e1 = A*x_1-b
```

School of AI, AVV

```
e1 = 3×1  
10-4 ×  
-0.2632  
0.0214  
-0.0122
```

```
e2 = A*x_2-b
```

```
e2 = 3×1  
10-13 ×  
-0.1066  
0.2842  
0
```

Check the sparsity of the solution

```
sum(x_1 ~= 0)
```

```
ans =  
3
```

```
sum(x_2 ~= 0)
```

```
ans =  
10
```

```
cd("/media/user/DATA4LINUX/new1/Repos/Mine/MFC4_22MAT230/")  
mlxfile = matlab.desktop.editor.getActive().Filename;  
outfile = mlxfile + ".pdf";  
export(matlab.desktop.editor.getActive().Filename, outfile, PageSize="A4");
```