

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 = 3×1  
-28  
21  
21
```

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;
```

Initialize the \mathbf{z} and \mathbf{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  
end
```

```

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 : 4.705226 dxz : 1.369830
iter : 2      dz : 0.796924 dxz : 3.478089
iter : 3      dz : 0.689179 dxz : 0.708165
iter : 4      dz : 0.371754 dxz : 0.402278
iter : 5      dz : 0.353023 dxz : 0.359279
iter : 6      dz : 0.196943 dxz : 0.206037
iter : 7      dz : 0.097210 dxz : 0.143178
iter : 8      dz : 0.089849 dxz : 0.115345
iter : 9      dz : 0.097634 dxz : 0.100401
iter : 10     dz : 0.091017 dxz : 0.092670
iter : 11     dz : 0.086026 dxz : 0.088781
iter : 12     dz : 0.085462 dxz : 0.086835
iter : 13     dz : 0.085202 dxz : 0.085847
iter : 14     dz : 0.084865 dxz : 0.085330
iter : 15     dz : 0.084863 dxz : 0.085048
iter : 16     dz : 0.084863 dxz : 0.084888
iter : 17     dz : 0.084734 dxz : 0.084794
iter : 18     dz : 0.084637 dxz : 0.084735
iter : 19     dz : 0.084631 dxz : 0.084698
iter : 20     dz : 0.084656 dxz : 0.084674
iter : 21     dz : 0.084658 dxz : 0.084658
iter : 22     dz : 0.084642 dxz : 0.084648
iter : 23     dz : 0.084628 dxz : 0.084640
iter : 24     dz : 0.084625 dxz : 0.084636
iter : 25     dz : 0.084628 dxz : 0.084632
iter : 26     dz : 0.084630 dxz : 0.084630
iter : 27     dz : 0.084628 dxz : 0.084628
iter : 28     dz : 0.084626 dxz : 0.084627
iter : 29     dz : 0.084625 dxz : 0.084627
iter : 30     dz : 0.084625 dxz : 0.084626
iter : 31     dz : 0.084626 dxz : 0.084626
iter : 32     dz : 0.084625 dxz : 0.084625
iter : 33     dz : 0.084625 dxz : 0.084625
iter : 34     dz : 0.084625 dxz : 0.084625
iter : 35     dz : 0.084625 dxz : 0.084625
iter : 36     dz : 0.084625 dxz : 0.084625
iter : 37     dz : 0.084625 dxz : 0.084625
iter : 38     dz : 0.084625 dxz : 0.084625
iter : 39     dz : 0.084625 dxz : 0.084625
iter : 40     dz : 0.084625 dxz : 0.084625
iter : 41     dz : 0.084625 dxz : 0.084625
iter : 42     dz : 0.084625 dxz : 0.084625
iter : 43     dz : 0.084625 dxz : 0.084625
iter : 44     dz : 0.084625 dxz : 0.084625
iter : 45     dz : 0.084625 dxz : 0.084625
iter : 46     dz : 0.084625 dxz : 0.084625
iter : 47     dz : 0.084625 dxz : 0.084625
iter : 48     dz : 0.084625 dxz : 0.084625
iter : 49     dz : 0.084625 dxz : 0.084625
iter : 50     dz : 0.082446 dxz : 0.082527
iter : 51     dz : 0.068452 dxz : 0.074566
iter : 52     dz : 0.052207 dxz : 0.069616
iter : 53     dz : 0.033211 dxz : 0.065407
iter : 54     dz : 0.013710 dxz : 0.061611
iter : 55     dz : 0.009272 dxz : 0.058110
iter : 56     dz : 0.024697 dxz : 0.054848
iter : 57     dz : 0.037039 dxz : 0.051794
iter : 58     dz : 0.044251 dxz : 0.048924
iter : 59     dz : 0.045959 dxz : 0.046222
iter : 60     dz : 0.042561 dxz : 0.043676

```

iter : 61	dz : 0.034992	dxz : 0.041274
iter : 62	dz : 0.024515	dxz : 0.039007
iter : 63	dz : 0.012557	dxz : 0.036866
iter : 64	dz : 0.000769	dxz : 0.034843
iter : 65	dz : 0.010301	dxz : 0.032932
iter : 66	dz : 0.018851	dxz : 0.031127
iter : 67	dz : 0.024485	dxz : 0.029420
iter : 68	dz : 0.026908	dxz : 0.027808
iter : 69	dz : 0.026204	dxz : 0.026283
iter : 70	dz : 0.022782	dxz : 0.024843
iter : 71	dz : 0.017296	dxz : 0.023481
iter : 72	dz : 0.010551	dxz : 0.022194
iter : 73	dz : 0.003401	dxz : 0.020978
iter : 74	dz : 0.003350	dxz : 0.019828
iter : 75	dz : 0.009024	dxz : 0.018742
iter : 76	dz : 0.013131	dxz : 0.017715
iter : 77	dz : 0.015401	dxz : 0.016744
iter : 78	dz : 0.015787	dxz : 0.015826
iter : 79	dz : 0.014449	dxz : 0.014959
iter : 80	dz : 0.011713	dxz : 0.014139
iter : 81	dz : 0.008020	dxz : 0.013364
iter : 82	dz : 0.003866	dxz : 0.012632
iter : 83	dz : 0.000257	dxz : 0.011940
iter : 84	dz : 0.003914	dxz : 0.011285
iter : 85	dz : 0.006763	dxz : 0.010667
iter : 86	dz : 0.008588	dxz : 0.010082
iter : 87	dz : 0.009302	dxz : 0.009530
iter : 88	dz : 0.008949	dxz : 0.009008
iter : 89	dz : 0.007680	dxz : 0.008514
iter : 90	dz : 0.005727	dxz : 0.008047
iter : 91	dz : 0.003372	dxz : 0.007606
iter : 92	dz : 0.000908	dxz : 0.007189
iter : 93	dz : 0.001389	dxz : 0.006795
iter : 94	dz : 0.003294	dxz : 0.006423
iter : 95	dz : 0.004645	dxz : 0.006071
iter : 96	dz : 0.005356	dxz : 0.005738
iter : 97	dz : 0.005421	dxz : 0.005424
iter : 98	dz : 0.004901	dxz : 0.005127
iter : 99	dz : 0.003914	dxz : 0.004846
iter : 100	dz : 0.002614	dxz : 0.004580
iter : 101	dz : 0.001175	dxz : 0.004329
iter : 102	dz : 0.000236	dxz : 0.004092
iter : 103	dz : 0.001471	dxz : 0.003868
iter : 104	dz : 0.002418	dxz : 0.003656
iter : 105	dz : 0.003007	dxz : 0.003455
iter : 106	dz : 0.003212	dxz : 0.003266
iter : 107	dz : 0.003052	dxz : 0.003087
iter : 108	dz : 0.002585	dxz : 0.002918
iter : 109	dz : 0.001891	dxz : 0.002758
iter : 110	dz : 0.001070	dxz : 0.002607
iter : 111	dz : 0.000223	dxz : 0.002464
iter : 112	dz : 0.000558	dxz : 0.002329
iter : 113	dz : 0.001196	dxz : 0.002201
iter : 114	dz : 0.001639	dxz : 0.002081
iter : 115	dz : 0.001860	dxz : 0.001967
iter : 116	dz : 0.001859	dxz : 0.001859
iter : 117	dz : 0.001660	dxz : 0.001757
iter : 118	dz : 0.001305	dxz : 0.001661
iter : 119	dz : 0.000849	dxz : 0.001570
iter : 120	dz : 0.000351	dxz : 0.001484
iter : 121	dz : 0.000131	dxz : 0.001402
iter : 122	dz : 0.000548	dxz : 0.001326
iter : 123	dz : 0.000862	dxz : 0.001253
iter : 124	dz : 0.001051	dxz : 0.001184
iter : 125	dz : 0.001107	dxz : 0.001119
iter : 126	dz : 0.001040	dxz : 0.001058
iter : 127	dz : 0.000869	dxz : 0.001000
iter : 128	dz : 0.000623	dxz : 0.000945

iter : 129	dz : 0.000337	dxz : 0.000893
iter : 130	dz : 0.000046	dxz : 0.000844
iter : 131	dz : 0.000219	dxz : 0.000798
iter : 132	dz : 0.000433	dxz : 0.000754
iter : 133	dz : 0.000577	dxz : 0.000713
iter : 134	dz : 0.000645	dxz : 0.000674
iter : 135	dz : 0.000637	dxz : 0.000637
iter : 136	dz : 0.000561	dxz : 0.000602
iter : 137	dz : 0.000434	dxz : 0.000569
iter : 138	dz : 0.000274	dxz : 0.000538
iter : 139	dz : 0.000102	dxz : 0.000508
iter : 140	dz : 0.000062	dxz : 0.000481
iter : 141	dz : 0.000203	dxz : 0.000454
iter : 142	dz : 0.000307	dxz : 0.000429
iter : 143	dz : 0.000367	dxz : 0.000406
iter : 144	dz : 0.000381	dxz : 0.000384
iter : 145	dz : 0.000354	dxz : 0.000363
iter : 146	dz : 0.000291	dxz : 0.000343
iter : 147	dz : 0.000205	dxz : 0.000324
iter : 148	dz : 0.000105	dxz : 0.000306
iter : 149	dz : 0.000005	dxz : 0.000289
iter : 150	dz : 0.000085	dxz : 0.000274
iter : 151	dz : 0.000156	dxz : 0.000259
iter : 152	dz : 0.000203	dxz : 0.000244
iter : 153	dz : 0.000223	dxz : 0.000231
iter : 154	dz : 0.000218	dxz : 0.000218
iter : 155	dz : 0.000190	dxz : 0.000206
iter : 156	dz : 0.000144	dxz : 0.000195
iter : 157	dz : 0.000088	dxz : 0.000184
iter : 158	dz : 0.000029	dxz : 0.000174
iter : 159	dz : 0.000027	dxz : 0.000165
iter : 160	dz : 0.000074	dxz : 0.000156
iter : 161	dz : 0.000109	dxz : 0.000147
iter : 162	dz : 0.000128	dxz : 0.000139
iter : 163	dz : 0.000131	dxz : 0.000131
iter : 164	dz : 0.000120	dxz : 0.000124
iter : 165	dz : 0.000098	dxz : 0.000117
iter : 166	dz : 0.000067	dxz : 0.000111
iter : 167	dz : 0.000032	dxz : 0.000105
iter : 168	dz : 0.000002	dxz : 0.000099
iter : 169	dz : 0.000032	dxz : 0.000094
iter : 170	dz : 0.000056	dxz : 0.000089
iter : 171	dz : 0.000071	dxz : 0.000084
iter : 172	dz : 0.000077	dxz : 0.000079
iter : 173	dz : 0.000074	dxz : 0.000075
iter : 174	dz : 0.000064	dxz : 0.000071
iter : 175	dz : 0.000048	dxz : 0.000067
iter : 176	dz : 0.000028	dxz : 0.000063
iter : 177	dz : 0.000008	dxz : 0.000060
iter : 178	dz : 0.000011	dxz : 0.000056
iter : 179	dz : 0.000027	dxz : 0.000053
iter : 180	dz : 0.000038	dxz : 0.000050
iter : 181	dz : 0.000044	dxz : 0.000048
iter : 182	dz : 0.000045	dxz : 0.000045
iter : 183	dz : 0.000041	dxz : 0.000043
iter : 184	dz : 0.000033	dxz : 0.000040
iter : 185	dz : 0.000022	dxz : 0.000038
iter : 186	dz : 0.000010	dxz : 0.000036
iter : 187	dz : 0.000002	dxz : 0.000034
iter : 188	dz : 0.000012	dxz : 0.000032
iter : 189	dz : 0.000020	dxz : 0.000030
iter : 190	dz : 0.000025	dxz : 0.000029
iter : 191	dz : 0.000027	dxz : 0.000027
iter : 192	dz : 0.000025	dxz : 0.000026
iter : 193	dz : 0.000022	dxz : 0.000024
iter : 194	dz : 0.000016	dxz : 0.000023
iter : 195	dz : 0.000009	dxz : 0.000022
iter : 196	dz : 0.000002	dxz : 0.000020

```

iter : 197      dz : 0.000005 dxz : 0.000019
iter : 198      dz : 0.000010 dxz : 0.000018
iter : 199      dz : 0.000014 dxz : 0.000017
iter : 200      dz : 0.000015 dxz : 0.000016
iter : 201      dz : 0.000015 dxz : 0.000015
iter : 202      dz : 0.000014 dxz : 0.000015
iter : 203      dz : 0.000011 dxz : 0.000014
iter : 204      dz : 0.000007 dxz : 0.000013
iter : 205      dz : 0.000003 dxz : 0.000012
iter : 206      dz : 0.000001 dxz : 0.000012
iter : 207      dz : 0.000005 dxz : 0.000011
iter : 208      dz : 0.000007 dxz : 0.000010
iter : 209      dz : 0.000009 dxz : 0.000010
iter : 210      dz : 0.000009 dxz : 0.000009
iter : 211      dz : 0.000009 dxz : 0.000009
iter : 212      dz : 0.000007 dxz : 0.000008
iter : 213      dz : 0.000005 dxz : 0.000008
iter : 214      dz : 0.000003 dxz : 0.000007

```

```

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

```

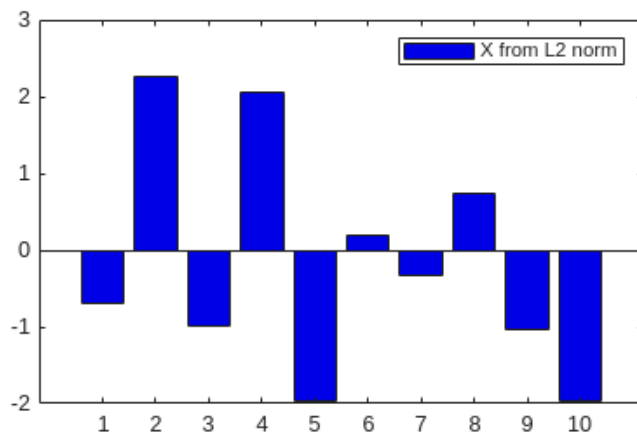
L1 norm optimized solution found after 215 iterations.

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")

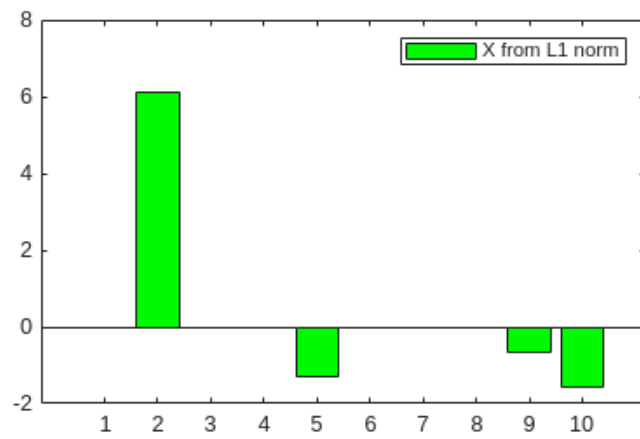
```



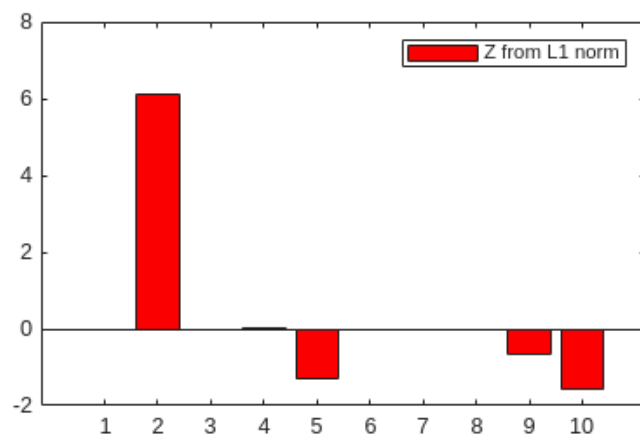
```

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
```

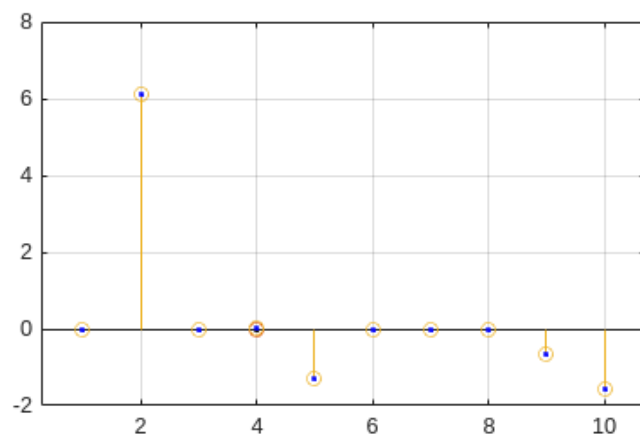
```
stem(X1)
```

```
stem(Z1)
```

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

```
hold off
```

```
grid on
```



Check the error vector $Ax - b$

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

```
e1 = 3×1  
10-4 ×  
    0.2364  
   -0.2869  
    0.6553
```

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

```
e2 = 3×1  
10-14 ×  
   -0.7105  
    0.7105  
         0
```

Check the sparsity of the solution

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

```
ans =  
4
```

```
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");
```

```
outfile =  
"/media/user/DATA4LINUX/new1/Repos/Mine/MFC4_22MAT230/U1_Comparison_L1_L2_norm_Matlab_coding.mlx.pdf"
```