

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  
-20  
10  
-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-5;
```

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
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 : 4.407867 dxz : 1.258358
iter : 2      dz : 0.857881 dxz : 2.327676
iter : 3      dz : 0.601731 dxz : 0.657268
iter : 4      dz : 0.133610 dxz : 0.184285
iter : 5      dz : 0.104715 dxz : 0.115283
iter : 6      dz : 0.077729 dxz : 0.090324
iter : 7      dz : 0.079818 dxz : 0.081613
iter : 8      dz : 0.077984 dxz : 0.078412
iter : 9      dz : 0.076242 dxz : 0.077144
iter : 10     dz : 0.076259 dxz : 0.076608
iter : 11     dz : 0.076353 dxz : 0.076370
iter : 12     dz : 0.076231 dxz : 0.076258
iter : 13     dz : 0.076145 dxz : 0.076201
iter : 14     dz : 0.076118 dxz : 0.076170
iter : 15     dz : 0.076109 dxz : 0.076152
iter : 16     dz : 0.076115 dxz : 0.076140
iter : 17     dz : 0.076124 dxz : 0.076131
iter : 18     dz : 0.076125 dxz : 0.076125
iter : 19     dz : 0.076117 dxz : 0.076120
iter : 20     dz : 0.076108 dxz : 0.076117
iter : 21     dz : 0.076104 dxz : 0.076114
iter : 22     dz : 0.076105 dxz : 0.076112
iter : 23     dz : 0.076107 dxz : 0.076110
iter : 24     dz : 0.076108 dxz : 0.076109
iter : 25     dz : 0.076108 dxz : 0.076108
iter : 26     dz : 0.076106 dxz : 0.076107
iter : 27     dz : 0.076104 dxz : 0.076106
iter : 28     dz : 0.076104 dxz : 0.076106
iter : 29     dz : 0.076104 dxz : 0.076105
iter : 30     dz : 0.076105 dxz : 0.076105
iter : 31     dz : 0.076105 dxz : 0.076105
iter : 32     dz : 0.076104 dxz : 0.076105
iter : 33     dz : 0.076104 dxz : 0.076104
iter : 34     dz : 0.076104 dxz : 0.076104
iter : 35     dz : 0.076104 dxz : 0.076104
iter : 36     dz : 0.076104 dxz : 0.076104
iter : 37     dz : 0.076104 dxz : 0.076104
iter : 38     dz : 0.076104 dxz : 0.076104
iter : 39     dz : 0.076104 dxz : 0.076104
iter : 40     dz : 0.076104 dxz : 0.076104
iter : 41     dz : 0.076104 dxz : 0.076104
iter : 42     dz : 0.076104 dxz : 0.076104
iter : 43     dz : 0.076104 dxz : 0.076104
iter : 44     dz : 0.076104 dxz : 0.076104
iter : 45     dz : 0.076104 dxz : 0.076104
iter : 46     dz : 0.076104 dxz : 0.076104
iter : 47     dz : 0.076104 dxz : 0.076104
iter : 48     dz : 0.076104 dxz : 0.076104
iter : 49     dz : 0.068449 dxz : 0.069018
iter : 50     dz : 0.058336 dxz : 0.063453
iter : 51     dz : 0.048683 dxz : 0.060777

```

iter : 52	dz : 0.036680	dxz : 0.058503
iter : 53	dz : 0.022835	dxz : 0.056427
iter : 54	dz : 0.008328	dxz : 0.054472
iter : 55	dz : 0.005717	dxz : 0.052606
iter : 56	dz : 0.018366	dxz : 0.050814
iter : 57	dz : 0.028940	dxz : 0.049087
iter : 58	dz : 0.036891	dxz : 0.047422
iter : 59	dz : 0.041863	dxz : 0.045814
iter : 60	dz : 0.043715	dxz : 0.044262
iter : 61	dz : 0.042532	dxz : 0.042762
iter : 62	dz : 0.038595	dxz : 0.041313
iter : 63	dz : 0.032350	dxz : 0.039914
iter : 64	dz : 0.024366	dxz : 0.038561
iter : 65	dz : 0.015291	dxz : 0.037255
iter : 66	dz : 0.005801	dxz : 0.035993
iter : 67	dz : 0.003442	dxz : 0.034774
iter : 68	dz : 0.011841	dxz : 0.033596
iter : 69	dz : 0.018891	dxz : 0.032458
iter : 70	dz : 0.024214	dxz : 0.031358
iter : 71	dz : 0.027569	dxz : 0.030296
iter : 72	dz : 0.028865	dxz : 0.029269
iter : 73	dz : 0.028152	dxz : 0.028278
iter : 74	dz : 0.025611	dxz : 0.027320
iter : 75	dz : 0.021534	dxz : 0.026394
iter : 76	dz : 0.016294	dxz : 0.025500
iter : 77	dz : 0.010317	dxz : 0.024636
iter : 78	dz : 0.004052	dxz : 0.023802
iter : 79	dz : 0.002066	dxz : 0.022996
iter : 80	dz : 0.007639	dxz : 0.022217
iter : 81	dz : 0.012332	dxz : 0.021464
iter : 82	dz : 0.015891	dxz : 0.020737
iter : 83	dz : 0.018154	dxz : 0.020034
iter : 84	dz : 0.019058	dxz : 0.019356
iter : 85	dz : 0.018632	dxz : 0.018700
iter : 86	dz : 0.016993	dxz : 0.018066
iter : 87	dz : 0.014332	dxz : 0.017454
iter : 88	dz : 0.010894	dxz : 0.016863
iter : 89	dz : 0.006959	dxz : 0.016292
iter : 90	dz : 0.002822	dxz : 0.015740
iter : 91	dz : 0.001227	dxz : 0.015207
iter : 92	dz : 0.004924	dxz : 0.014692
iter : 93	dz : 0.008048	dxz : 0.014194
iter : 94	dz : 0.010427	dxz : 0.013713
iter : 95	dz : 0.011953	dxz : 0.013249
iter : 96	dz : 0.012582	dxz : 0.012800
iter : 97	dz : 0.012330	dxz : 0.012366
iter : 98	dz : 0.011274	dxz : 0.011947
iter : 99	dz : 0.009538	dxz : 0.011542
iter : 100	dz : 0.007282	dxz : 0.011151
iter : 101	dz : 0.004691	dxz : 0.010774
iter : 102	dz : 0.001960	dxz : 0.010409
iter : 103	dz : 0.000719	dxz : 0.010056
iter : 104	dz : 0.003172	dxz : 0.009715
iter : 105	dz : 0.005251	dxz : 0.009386
iter : 106	dz : 0.006841	dxz : 0.009068
iter : 107	dz : 0.007869	dxz : 0.008761
iter : 108	dz : 0.008305	dxz : 0.008464
iter : 109	dz : 0.008159	dxz : 0.008178
iter : 110	dz : 0.007479	dxz : 0.007901
iter : 111	dz : 0.006347	dxz : 0.007633
iter : 112	dz : 0.004866	dxz : 0.007374
iter : 113	dz : 0.003161	dxz : 0.007125
iter : 114	dz : 0.001358	dxz : 0.006883
iter : 115	dz : 0.000415	dxz : 0.006650

iter : 116	dz : 0.002042	dxz : 0.006425
iter : 117	dz : 0.003425	dxz : 0.006207
iter : 118	dz : 0.004487	dxz : 0.005997
iter : 119	dz : 0.005180	dxz : 0.005794
iter : 120	dz : 0.005482	dxz : 0.005597
iter : 121	dz : 0.005399	dxz : 0.005408
iter : 122	dz : 0.004961	dxz : 0.005225
iter : 123	dz : 0.004223	dxz : 0.005048
iter : 124	dz : 0.003252	dxz : 0.004877
iter : 125	dz : 0.002129	dxz : 0.004711
iter : 126	dz : 0.000939	dxz : 0.004552
iter : 127	dz : 0.000234	dxz : 0.004398
iter : 128	dz : 0.001313	dxz : 0.004249
iter : 129	dz : 0.002233	dxz : 0.004105
iter : 130	dz : 0.002943	dxz : 0.003966
iter : 131	dz : 0.003410	dxz : 0.003831
iter : 132	dz : 0.003618	dxz : 0.003702
iter : 133	dz : 0.003572	dxz : 0.003576
iter : 134	dz : 0.003291	dxz : 0.003455
iter : 135	dz : 0.002809	dxz : 0.003338
iter : 136	dz : 0.002172	dxz : 0.003225
iter : 137	dz : 0.001433	dxz : 0.003116
iter : 138	dz : 0.000648	dxz : 0.003010
iter : 139	dz : 0.000128	dxz : 0.002908
iter : 140	dz : 0.000844	dxz : 0.002810
iter : 141	dz : 0.001456	dxz : 0.002714
iter : 142	dz : 0.001930	dxz : 0.002622
iter : 143	dz : 0.002244	dxz : 0.002534
iter : 144	dz : 0.002388	dxz : 0.002448
iter : 145	dz : 0.002363	dxz : 0.002365
iter : 146	dz : 0.002182	dxz : 0.002285
iter : 147	dz : 0.001868	dxz : 0.002207
iter : 148	dz : 0.001451	dxz : 0.002133
iter : 149	dz : 0.000965	dxz : 0.002060
iter : 150	dz : 0.000446	dxz : 0.001991
iter : 151	dz : 0.000067	dxz : 0.001923
iter : 152	dz : 0.000542	dxz : 0.001858
iter : 153	dz : 0.000949	dxz : 0.001795
iter : 154	dz : 0.001265	dxz : 0.001734
iter : 155	dz : 0.001477	dxz : 0.001675
iter : 156	dz : 0.001576	dxz : 0.001619
iter : 157	dz : 0.001563	dxz : 0.001564
iter : 158	dz : 0.001447	dxz : 0.001511
iter : 159	dz : 0.001243	dxz : 0.001460
iter : 160	dz : 0.000969	dxz : 0.001410
iter : 161	dz : 0.000649	dxz : 0.001362
iter : 162	dz : 0.000307	dxz : 0.001316
iter : 163	dz : 0.000033	dxz : 0.001272
iter : 164	dz : 0.000347	dxz : 0.001229
iter : 165	dz : 0.000618	dxz : 0.001187
iter : 166	dz : 0.000830	dxz : 0.001147
iter : 167	dz : 0.000972	dxz : 0.001108
iter : 168	dz : 0.001040	dxz : 0.001070
iter : 169	dz : 0.001034	dxz : 0.001034
iter : 170	dz : 0.000960	dxz : 0.000999
iter : 171	dz : 0.000826	dxz : 0.000965
iter : 172	dz : 0.000647	dxz : 0.000933
iter : 173	dz : 0.000436	dxz : 0.000901
iter : 174	dz : 0.000211	dxz : 0.000870
iter : 175	dz : 0.000014	dxz : 0.000841
iter : 176	dz : 0.000223	dxz : 0.000812
iter : 177	dz : 0.000403	dxz : 0.000785
iter : 178	dz : 0.000544	dxz : 0.000758
iter : 179	dz : 0.000639	dxz : 0.000733

```

iter : 180      dz : 0.000686 dxz : 0.000708
iter : 181      dz : 0.000684 dxz : 0.000684
iter : 182      dz : 0.000636 dxz : 0.000661
iter : 183      dz : 0.000549 dxz : 0.000638
iter : 184      dz : 0.000432 dxz : 0.000617
iter : 185      dz : 0.000293 dxz : 0.000596
iter : 186      dz : 0.000145 dxz : 0.000576

```

```

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

```

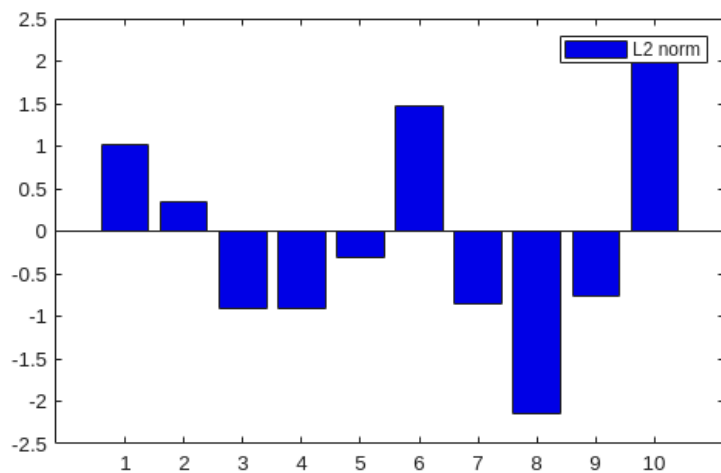
L1 norm optimized solution found after 187 iterations.

Plot to check the solution vector X and surrogate variable Z

```

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

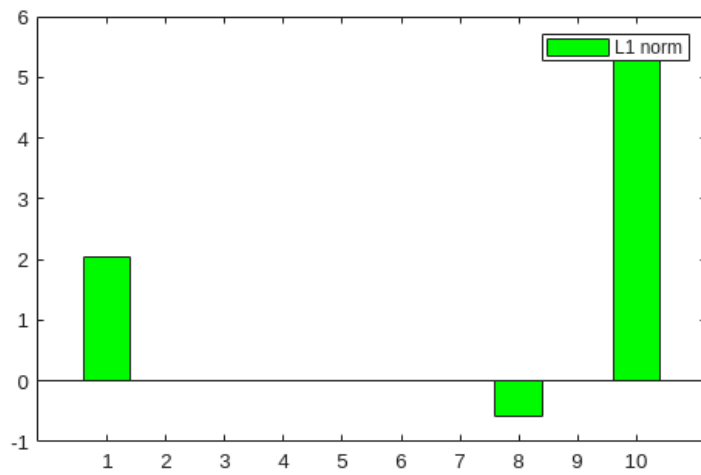
```



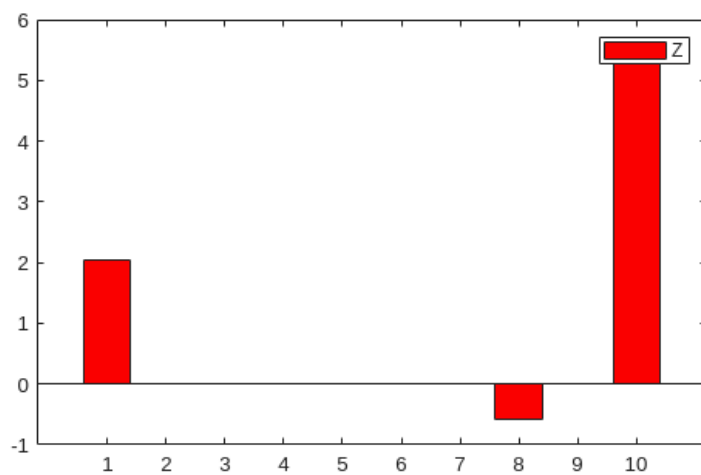
```

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

```



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



Check the error vector $Ax - b$

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

```
ans = 3×1
    0.0017
    0.0041
    0.0010
```

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

```
ans = 3×1
10-14 ×
    0.3553
```

0.3553
0.3553

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);
```