Module 07 – Assignment

Modern Navigation Systems – EN.525.645.81

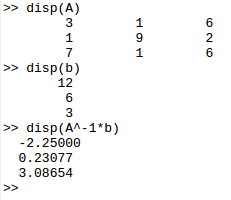
Submitted: 10/22/2018

**Kyle Mercer**

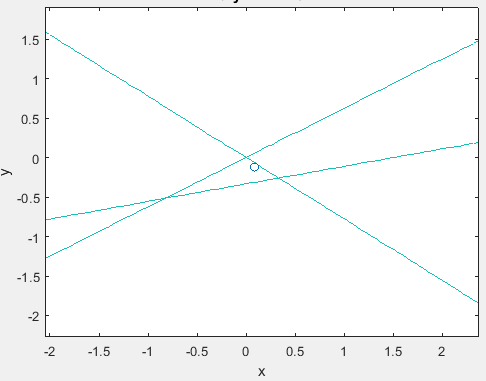
1. Solve a 3 x 3 system of simultaneous linear equations that you invent, using Cramer’s rule by hand. If you know how to invert a matrix by hand, or if you have the software that is capable of matrix inversion, use that technique as well. Then, check your results, by hand.

Cramer’s Method

Checking answer using MATLAB.



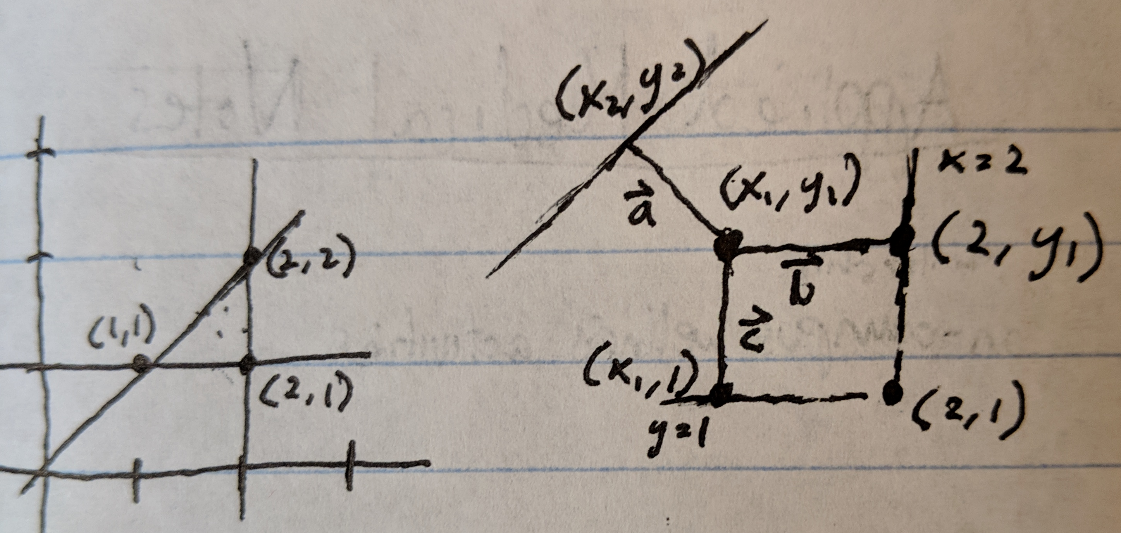
1. Invent a system of three equations in two unknowns. Graph the equations and find the least squares solution. Plot the solution on your graph.



Plot generated from the following MATALB code. Note matrix A and vector b were randomly generated. This function may be rerun to generate a new set of random functions with coefficients between -10 and 10.

|  |
| --- |
| coefr = [-10,10];  A = randi(coefr,3,2);  b = randi(coefr,3,1);    prange = [-10,10];  %%%%%%%%%%%%%%%%%%%%%%%  syms x y    f(x,y) = A(1,1)\*x + A(1,2)\*y - b(1);  g(x,y) = A(2,1)\*x + A(2,2)\*y - b(2);  h(x,y) = A(3,1)\*x + A(3,2)\*y - b(3);    fimplicit(f, prange);  hold on  fimplicit(g, prange);  fimplicit(h, prange);    u = (A'\*A)^-1\*A'\*b;    scatter(u(1),u(2));  hold off |

1. Using the example in the presentation, compute the coordinates x1, y1 by differentiating the expression for the sum of the squares of the distances to each of the three lines.



The line perpendicular to y = x is determined by inspection to be:

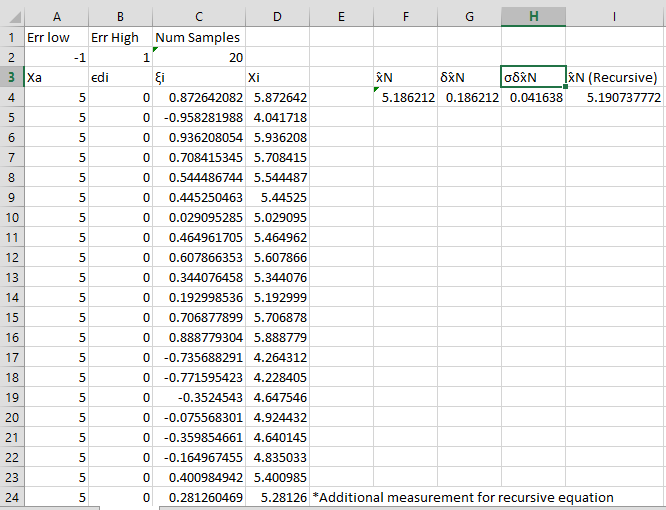
Solving for x2 and y2 by plugging in x for y:

Expressions for the distances of the vectors a, b, and c are:

Taking the derivatives of each function with respect to x1 and y1:

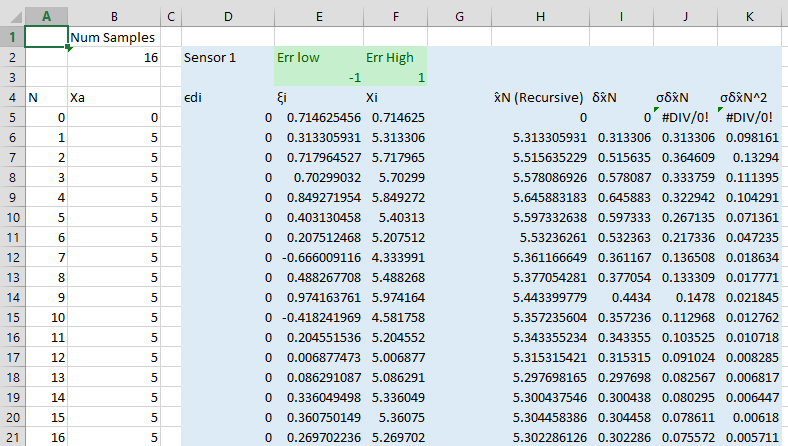
Solve the system of equations using the same least squares formula:

1. Using random numbers, and setting the deterministic error to zero, simulate equations 3.1 through 3.6 of the text.

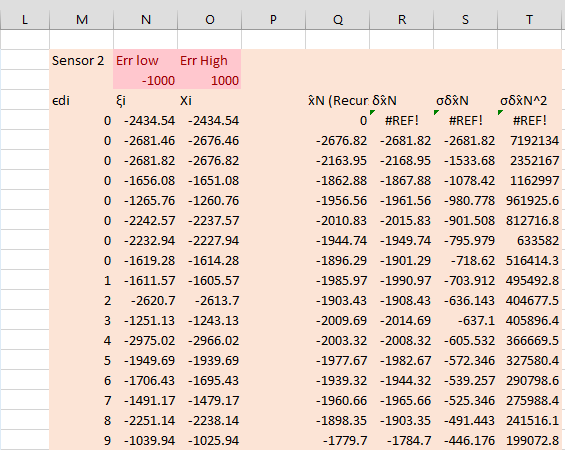


1. Extend this to simulation of equations 3.7 to 3.11. Make the simulated errors in one set of measurements significantly larger than those in the other, thus illustrating that inclusion of measurements with low weights is in principle helpful, but in practice is not always worth doing.

Sensor 1 data (with error from -1 to 1)

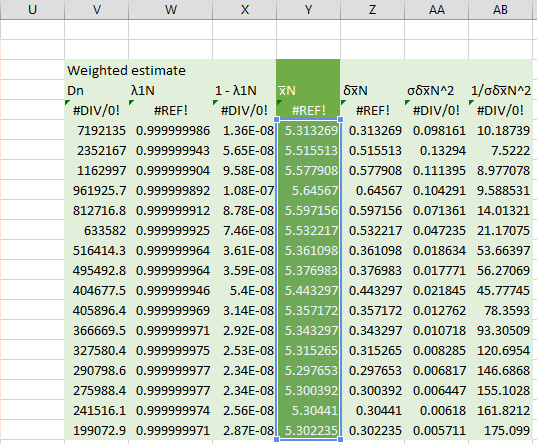


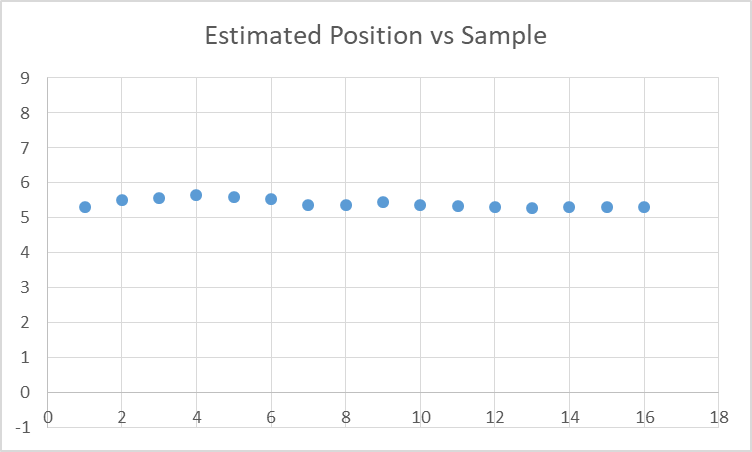
Sensor 2 data (with much greater error range from -1000 to 1000)



Weighted Estimate

(very close to true position of 5 despite sensor 2’s noise)





1. Comment on the relationship of the weighted least squares solution of an over-specified problem to the signal averaging technique developed in problems 1 and 2, and in chapter 3 of the primary text.

The distances calculated in least squares solution to an over-specified problem is similar to the signal averaging techniques because in both cases the goal is to come up with the best possible solution based upon the available data input to the system. In both processes, the error is calculated and used to obtain the optimal answer. In the case of the least squares solution, this error is considered the distance from the true answer to a point along an equation, whereas in the signal averaging approach this is the error introduced by random noise in the system and the subsequent calculation of variance among the data to determine how much weight (or how much trust) we should have in that sensor. The sensor is loosely analogous to an equation in the over-specified problem, however, the equation is perfectly linear whereas the sensor is noisy.