

The 3-point Problem
as Simultaneous Linear Equations
***** MatLab version *****

by Will Frangos

The famous three-point problem is important in structural interpretation. The idea is to determine the strike and dip of a stratum from three points, such as might be encountered in drill holes. The underlying assumption is that the horizon in question is a plane. Graphical methods of solution are commonly employed, lending an intuitive understanding and promoting three-dimensional visualization.

An analytic geometry approach has certain advantages over graphical solution, including precision and rapidity, albeit at the expense of intuitive visualization of the problem. Put succinctly, a mathematical solution would be very handy if you've got 50 of them to do; who wants to perform 50 graphical analyses before dinner?

The method is actually quite straightforward. First, we need to be able to describe an inclined plane by a mathematical function. There are several ways to do it, but let's use a simple slope-intercept form in a standard Cartesian coordinate system;

$$z = m_x x + m_y y + z_0$$

where x , y , and z are the coordinates,

m_x, m_y are the slopes in the x and y directions, Eqn 1

and z_0 is the elevation of the plane at the origin

i.e., the zero intercept

By this expression we mean that any point on the given plane at lateral location (x,y) will have an elevation of z . The tricky bit, of course, is to figure out the values of the slopes and the zero intercept.

Well, if we are given three points on the plane, such as the observed depths in a set of three drill holes of known locations, then we can form a set of simultaneous linear equations and figure out the slopes and zero intercept. Once we know them, we can then predict the elevation at which to expect the horizon at any other location. We can also figure out the strike and dip from the parameters.

The equations look like this:

$$z_1 = m_x x_1 + m_y y_1 + z_0$$

$$z_2 = m_x x_2 + m_y y_2 + z_0$$

$$z_3 = m_x x_3 + m_y y_3 + z_0$$

Eqn 2

and we want to determine m_x , m_y , and z_0 . Note that x, y , and z are *not* the unknowns here! They are the coordinates in this problem.

We could solve the matter for any set of three points by junior high school methods, but maybe there's a better way using computers. To do that, let's state the set of equations in matrix form, like this:

$$\bar{z} = \bar{A} \bar{u}$$

where \bar{z} is a column vector of the observed depths,

\bar{A} is the matrix of coefficients, x_n, y_n , and 1 (for the z_0),

and \bar{u} is the vector of unknown parameters to be found, m_x, m_y , and z_0

Eqn 3

The solution that we need is then

$$\bar{u} = \bar{A}^{-1} \bar{z}$$

where \bar{A}^{-1} is the inverse of matrix \bar{A}

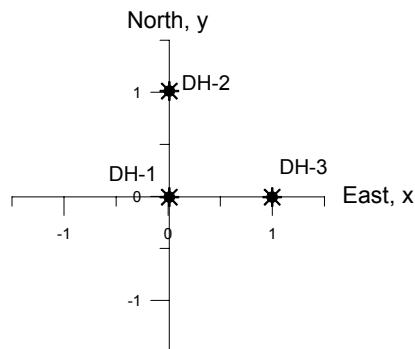
Eqn 4

This gets us into the realm of many very handy computer routines, and thus, easy sailing.

In practice, we won't actually calculate the matrix inverse of \bar{A} ; it turns out to be tedious and sometimes inaccurate, even for modern computers. But there are many good, reliable, accurate, fast (not that we worry about speed at this stage), and well-proven algorithms for solving systems of linear equations like this one. One alternative is to use program *LINEQ*, a linear equation solver available on our computer lab machines. Another more general one is to use *MATLAB*, a good, high-level, easily learned mathematics application which is made for matrices. We'll develop the latter in our examples.

Examples

Let's start with a simple, easily visualized case, as sketched below:



The horizontal coordinates for the drill holes are $(0,0)$, $(0,1)$, and $(1,0)$, respectively. Recalling that the coefficient of z_0 is 1 in the general equations, Eqn 2 above, the matrix becomes

$$\bar{A} = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix}$$

The simplest case is a plane with no dip at all, *i.e.*, all the depths are the same value. Suppose that depth is 6; then the column vector, \bar{z} , is a column of 6's. Let's solve the system and see if we get the right answers.

First, start up MatLab and enter the matrix, perhaps like this:

```
>> A=[0,0,1; 0,1,1; 1,0,1]    (the commas separate individual matrix elements,
                                the semicolons indicate new rows, and the square
                                brackets delineate the whole matrix)
```

```
MatLab replies:      A =
                     0    0    1
                     0    1    1
                     1    0    1    verifying that we got it right
```

Now enter the column vector of elevations:

```
>> z=[6;6;6]
```

```
again, MatLab replies:      z =
                             6
                             6
                             6
```

Now, to solve the system of simultaneous linear equations. In MatLab, this is conveniently indicated as a sort of division (even though it isn't really; there isn't any such thing as division in matrix algebra). Enter the following line:

```
>> u=A\z    Note the backslash, which causes the right stuff to happen, and the order!
```

```
MatLab replies:      u =
                     0      this is  $m_x$ 
                     0      this is  $m_y$ 
                     6      and this is  $z_0$ 
```

That's what we expected: no slope in the x - or y -directions, and a zero intercept of 6. Guess we must have got it right.

Exercise: Test the method a little further by specifying a new set of elevations. Try $\bar{z} = [6;6;7]$, corresponding to a plane that dips east with slope of 1. You should get $u = [1,0,6]$. Now try a plane dipping northeast with dips of 1 in each of the north and east directions.

Translation back to Geology:

Being geologists, we'd like to have the plane described as a stratum or horizon with strike and dip, not slope and intercept. We thank the mathematicians for their assistance and proceed as follows:

For the dip, we resolve the two slopes as vectors and convert to angles in degrees. By hand, or using a calculator, we calculate

$$\text{Total Slope} = \sqrt{m_x^2 + m_y^2}$$

and

$$\text{Dip}, \theta = \tan^{-1}(\text{slope})$$

and try to remember to use degrees, not radian, mode.

Assuming that we've still got MatLab running, we can get the dip and strike using the square root and the arctan functions. The dip can be done in two MatLab statements or one, depending on personal preference. Here is the northeast-dipping example done as one statement:

```
>> dip=180/pi * atan(sqrt(1*1+1*1))
```

MatLab replies: dip =
 54.7356 (the 180/pi converts to degrees)

To get the strike direction, we'll use some common sense and the four-quadrant arctan function. We want the direction from north (not east as the mathematicians do it), so we interchange the x- and y-components from what MatLab specifies, *i.e.*, we enter $\text{atan2}(m_y, m_x)$. For the last exercise above we would enter

```
>> dip=180/pi*atan2(1,1)
```

and MatLab replies dip =
 45

which is the direction of true dip, corresponding to northeast. The strike direction is thus northwest-southeast, so here's where the common sense comes in. We can add or subtract 90° , as appropriate, to express the strike direction.

Check this method to see if you get the right dip and direction for some other easy cases. In particular, set up some beds dipping to the southeast and southwest to illustrate the need for common sense in interpreting the results.