Using Excel's Solver - simple demo

Excel's Solver uses nonlinear least squares to evaluate an objective function during iterative convergence to a solution. Although there are quicker ways to find a least squares solution to a set of linear data doing so with Solver makes a quick demonstration of how to use it.

Step 1- Generate some "data" with a known slope and intercept for the linear equation:

Y = m*X + b, with

m = 0.777

b = 8.5

The blue diamonds (Figure 1) are the generated data from column "B". The red boxes are the generated results with a bit of noise added (column "C"). We'll use these noisy results along with Solver to extract the best fit model parameters (slope and intercept).

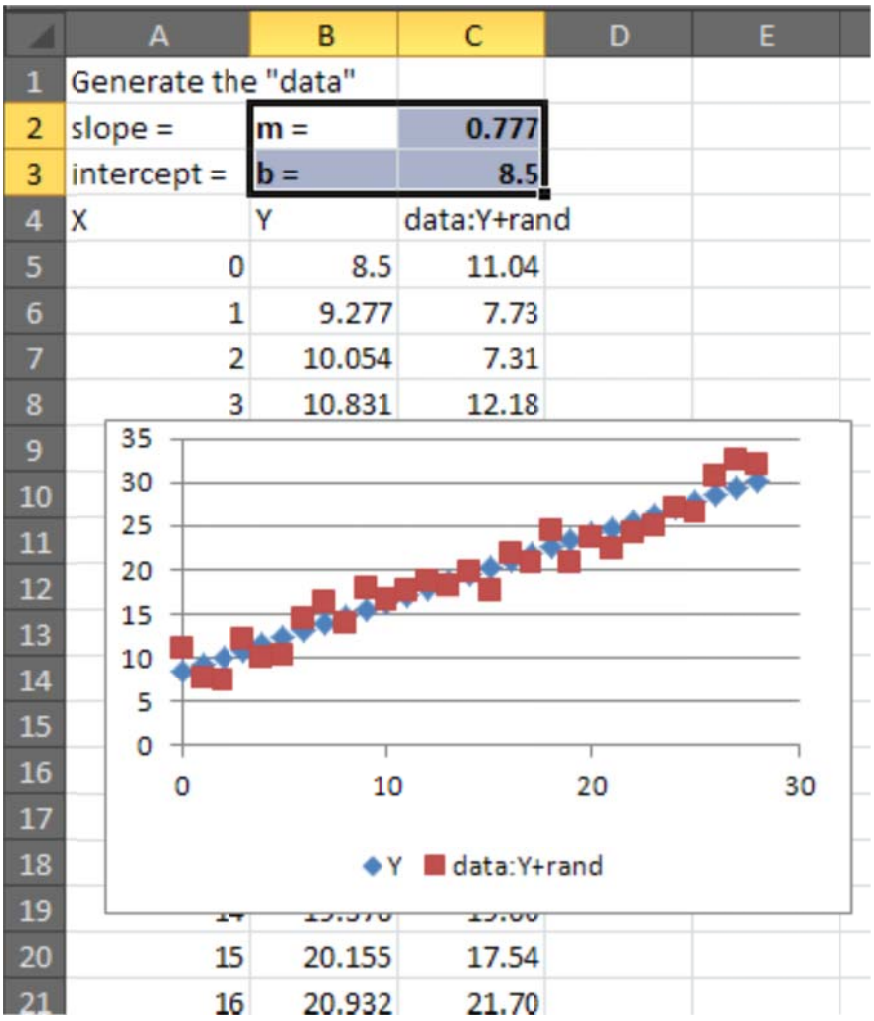

Figure 1. Initial spreadsheet to generate a column of results from a linear equation. Column one is X, column two is Y, and column three is Y with some random noise added to it. The graph shows the result.

Step 2 – Setup the fabricated data and a bad initial guess for slope and intercept

Here, the red boxes are the scattered data and the blue line is the fit from the bad guess (slope=1, intercept=2). Next, we use Solver to find the best fit parameters for the linear equation (Y = m*X + b); we expect to extract values close to those used to generate the "data".

Copy & paste "data" as values

		Guess:	slope =	1
			Intercept =	2
X	data:Y+rand	expected	difference^2	
0	7.56	2	30.8898076	
1	8.61	3	31.4459395	
2	8.07	4	16.5833004	
3	10.95	5	35.4577377	
17	22.41	19	11.6379518	

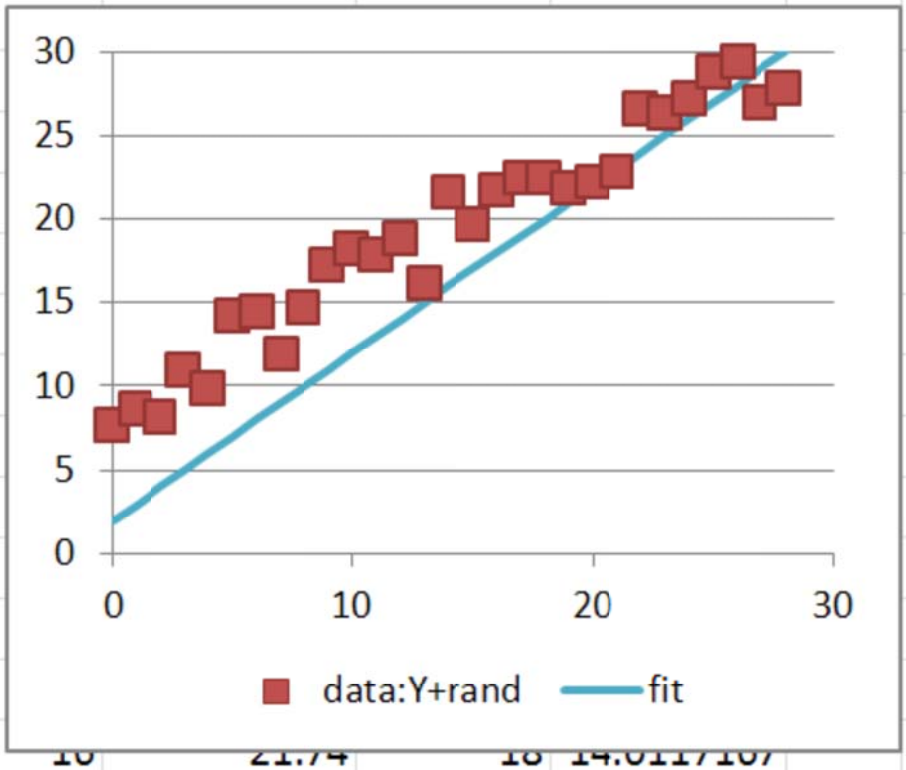

Figure 2. The next four columns contain X, the noisy data, initial results calculated for the guessed values of slope and intercept, and finally the squared difference between the data and the initial calculated values. The graph shows the poor fit of the guess to the data.

Step 3 – Use Solver to find the best fit parameters of the linear equation

The objective function to minimize is the sum of the squared differences between the data and the numbers calculated using the bad guess {=SQRT(SUM(J5:J33))}. That is, we make an initial guess of slope and intercept then use those values to calculate expectations. Following that, we difference the calculated and observed results and square that value. We put the square root of the sum of the squared difference in the objective cell (N4). Finally, we tell Solver where the objective cell is (N4) and tell it to minimize that value by iteratively changing cells K2 and K3 (slope and intercept). Solver will iteratively change the slope and intercept until the objective function is minimized - hit **"Solve"**.

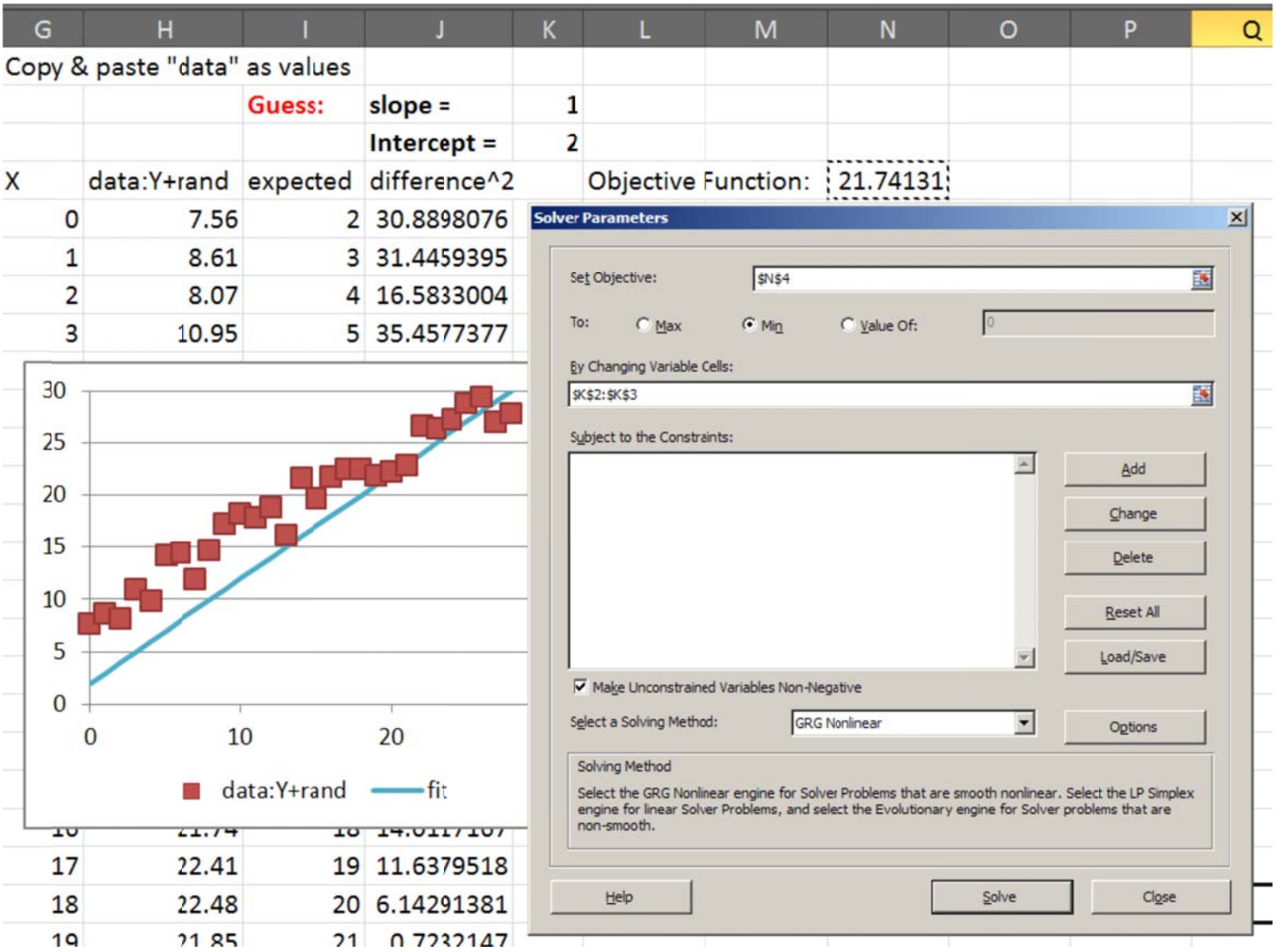

Figure 3. "Set Objective" is the box in which you point at the objective function you want to minimize (or possibly maximize). The "Changing Variable" cells hold the guessed slope and intercept. Solver updates these cells with iterative improvement of the guess until the objective function is minimized.

Step 4 – Look at the results from Solver

Solver recovered:

Slope = 0.76 (the original was 0.777)

Intercept = 8.58 (the original was 8.5)

Not bad.

Copy & paste "data" as values

		Guess:	slope =	0.76			
			Intercept =	8.58			
X	data:Y+rand	expected	difference^2		Objective Function:		7.763409
0	7.56	8.58185	1.04855598				
1							
2							
3							
4							
5							
6							
7							
8							
9							
10							
11							
12							
13	16.15	18.41724	5.12065752				
14	21.54	19.1738	5.58262552				
15	19.65	19.93037	0.07688163				

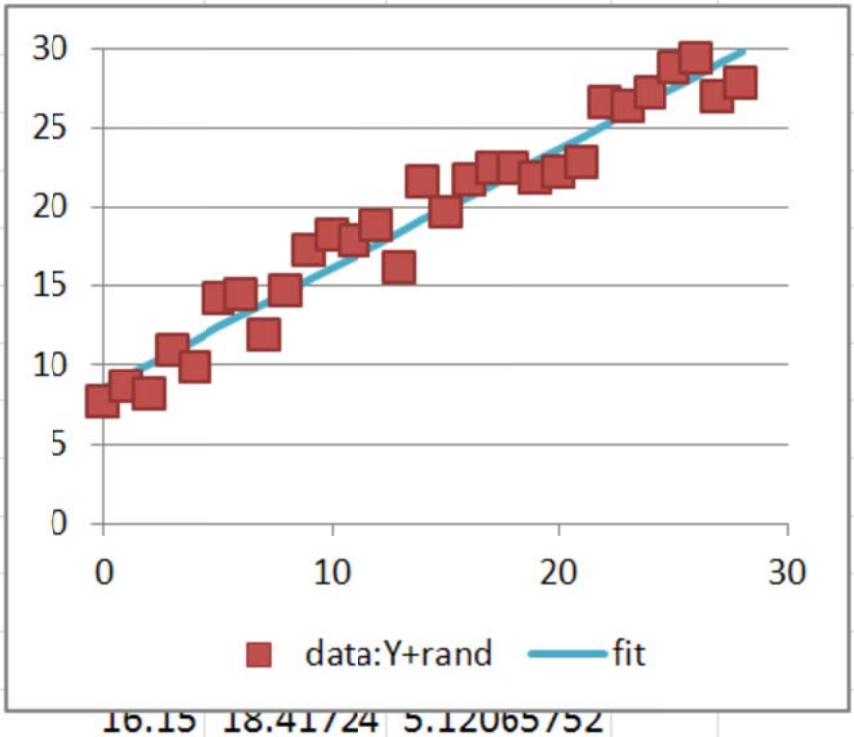

Figure 4. The blue line shows the fit of Solver's result to the noisy data shown by the red boxes.