

How to Easily Solve Systems of Equations in Excel

Authored by
stats writer

December 3, 2025

RECOMMENDED CITATION

stats writer (2025). *How to Easily Solve Systems of Equations in Excel*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=104269>

The process of solving a system of equations using Microsoft Excel offers several powerful methodologies, catering to users ranging from beginners to advanced analysts. While specialized software is often employed for massive datasets, Excel provides accessible tools for tackling linear algebra problems directly within the worksheet environment. One initial approach involves utilizing the built-in Solver tool, an optional add-in designed primarily for optimization and non-linear equation solving. To employ the Solver, users must first structure their linear equations directly into the worksheet, defining the relationships between the variables and setting up constraints.

After defining the structure, the user accesses the Solver tool found under the Data tab. Within the Solver Parameters dialog box, the core task involves setting the objective function (usually setting an equation result cell to a specific value, such as zero) and specifying the cells that hold the variables--referred to as the changing cells. For instance, if you had the equations $x+y=7$ and $x-y=3$, you would enter representations of these equations into cells A1 and B1, respectively, and designate cells A2 and B2 as the changing cells (for x and y). Upon clicking Solve, the Solver tool iteratively calculates the solution, populating the variable cells ($A2=5$, $B2=2$). However, for purely linear systems, the matrix method, utilizing Excel's powerful array functions, is often faster and yields exact results without relying on iteration.

For achieving robust and definitive solutions to linear systems, experts typically prefer Excel's native matrix manipulation functions. This sophisticated technique translates the set of linear equations into a standard matrix format, allowing Excel to apply established principles of linear algebra, specifically involving matrix inversion and multiplication. This method relies heavily on two primary functions: the MMULT (Matrix Multiplication) and the MINVERSE (Matrix Inverse) functions. The following examples demonstrate how to harness these powerful array functions to solve various systems of equations efficiently.

To solve a system of equations in Excel using linear algebra, we leverage the capabilities of the **MMULT** and **MINVERSE** functions. This technique is grounded in the matrix representation of linear systems ($AX = B$), where the solution X is derived by calculating $X = A^{-1}B$. The MINVERSE function computes the inverse of the coefficient matrix (A^{-1}), and the MMULT function then performs the multiplication of A^{-1} by the constant matrix B .

The following practical examples demonstrate the step-by-step process of setting up and solving systems ranging from two to four variables, providing a robust computational method for linear equation analysis.

Prerequisite Knowledge: Structuring Data for Matrix Operations

Before applying the matrix functions, meticulous data organization is essential. Every linear system must be accurately transformed into two distinct matrices: the coefficient matrix (A) and the constant matrix (B). The coefficient matrix A must be a square matrix ($N \times N$), where N is the

number of variables, ensuring that coefficients are placed in corresponding rows and columns relative to their original equations and variable order.

For instance, if solving for x, y, and z, the coefficients for x must occupy the first column, y the second, and z the third. If an equation lacks a variable, its coefficient must be explicitly represented as zero (0) in the input cell. The constant matrix B must be an Nx1 column vector, containing the right-hand side values of the equations, maintaining the same row order as the corresponding equations in matrix A.

This systematic arrangement not only facilitates accurate formula referencing but also prevents errors related to incompatible matrix dimensions during the MMULT calculation. The final output range selected for the solution matrix (X) must also be an Nx1 column vector, matching the size of the constant matrix B.

Example 1: Solving a System of Equations with Two Variables

Suppose we have the following system of equations and we'd like to solve for the values of x and y, demonstrating a foundational 2x2 matrix solution:

$$5x + 4y = 35$$

$$2x + 6y = 36$$

To solve this system of equations, we first structure the coefficients (5, 4, 2, 6) into a 2x2 coefficient matrix A (range A1:B2) and the constants (35, 36) into a 2x1 constant matrix B (range C1:C2) in our Excel sheet.

	A	B	C	D	E	F	G
1	5	4	35				
2	2	6	36				
3							
4							
5							
6							
7							
8							
9							
10							
11							
12							
13							
14							
15							

We can then use the following nested array formula, which calculates the product of the inverse of the coefficient matrix (MINVERSE) and the constant matrix (MMULT), to solve for the values of x and y:

=MMULT(MINVERSE(A1:B2),C1:C2)

We must select the two target cells (e.g., E1 and E2) to hold the solutions, type this formula into the formula bar, and then press **CTRL + SHIFT + ENTER** to execute it as an array formula, returning the solution matrix X:

	A	B	C	D	E	F	G
1	5	4	35		3		
2	2	6	36		5		
3							
4							
5							
6							
7							
8							
9							
10							
11							
12							
13							

The results in cells E1 and E2 demonstrate that the value for x is **3** and the value for y is **5**.

Example 2: Scaling Up to Three Variables (3x3 Matrix)

The matrix method is highly scalable. Let us now examine a system of equations involving three variables: x, y, and z. The principle remains the same, but the coefficient matrix A becomes 3x3, and the solution vector X becomes 3x1.

Suppose we have the following three-variable system and we'd like to solve for the values of x, y, and z:

$$4x + 2y + 1z = 34$$

$$3x + 5y - 2z = 41$$

$$1x + 1y + 3z = 21 \text{ (Assuming a complete system based on the matrix size)}$$

To solve this system, we first enter the coefficients into the 3x3 matrix A (range A1:C3) and the constants into the 3x1 matrix B (range D1:D3) in Excel:

	A	B	C	D	E	F	G
1	4	2	1	34			
2	3	5	-2	41			
3	2	2	4	30			
4							
5							
6							
7							
8							
9							
10							
11							
12							
13							
14							
15							

We then use the following expanded formula to solve for the values of x, y, and z, adjusting the ranges to accommodate the larger matrix size. We are calculating $X = \text{MINVERSE}(A1:C3) * D1:D3$:

=MMULT(MINVERSE(A1:C3),D1:D3)

We must select three contiguous output cells (e.g., F1 through F3) for the solution vector X, enter the formula into cell F1, and then press **CTRL + SHIFT + ENTER** to execute the array formula:

	A	B	C	D	E	F	G
1	4	2	1	34		5	
2	3	5	-2	41		6	
3	2	2	4	30		2	
4							
5							
6							
7							
8							
9							
10							
11							
12							
13							
14							
15							

The output confirms that the value for x is **5**, the value for y is **6**, and the value for z is **2**.

Understanding Array Formula Entry

It is essential to understand why the array entry (CTRL + SHIFT + ENTER) is required. Both MINVERSE and MMULT are array functions designed to return a range of values, not just a single cell result. Since the solution X contains a value for every variable, we need the formula to spill its output across multiple cells.

In older versions of Excel, failure to select the full output range and confirm the entry using CTRL + SHIFT + ENTER will result in only the first value of the solution matrix appearing, often accompanied by #VALUE! errors in adjacent cells or simply an incomplete solution. The curly braces {} automatically placed around the formula by Excel indicate a successful array entry.

Example 3: Solving Complex 4-Variable Systems (4x4 Matrix)

The matrix method scales efficiently, allowing us to solve systems with four or more variables without significant change to the methodology. For a four-variable system (w, x, y, z), we require a 4x4 coefficient matrix A and a 4x1 constant matrix B.

Suppose we have the following system of equations:

$$6w + 2x + 2y + 1z = 37$$

$$2w + 1x + 1y + 0z = 14$$

$$3w + 2x + 2y + 4z = 28$$

$$2w + 0x + 5y + 5z = 28$$

To solve this system of equations, we first type in the coefficients and constants, ensuring that coefficients of zero (0) are included where a variable is absent in an equation. The coefficient matrix A spans A1:D4, and the constant matrix B spans E1:E4:

	A	B	C	D	E	F	G
1	6	2	2	1	37		
2	2	1	1	0	14		
3	3	2	2	4	28		
4	2	0	5	5	28		
5							
6							
7							
8							
9							
10							
11							
12							
13							
14							
15							

We then use the following array formula to solve for the values of w, x, y, and z. This formula executes the calculation $\mathbf{X} = \mathbf{A}^{-1}\mathbf{B}$ using the expanded ranges:

=MMULT(MINVERSE(A1:D4),E1:E4)

We must select four cells (e.g., G1 through G4) and press **CTRL + SHIFT + ENTER**. The solution matrix X is then populated:

	A	B	C	D	E	F	G	H
1	6	2	2	1	37		4	
2	2	1	1	0	14		3	
3	3	2	2	4	28		3	
4	2	0	5	5	28		1	
5								
6								
7								
8								
9								
10								
11								
12								
13								
14								
15								
16								

This tells us that the value for w is **4**, x is **3**, y is **3**, and z is **1**, confirming the scalability of the matrix inversion method for solving high-dimensional systems in Excel.

Troubleshooting Common Errors

When using matrix functions, two primary errors can halt the calculation: #NUM! and #VALUE!. A #NUM! error usually indicates that the coefficient matrix A is singular (non-invertible), meaning its determinant is zero. This mathematical condition arises when the system of equations has either no unique solution or infinitely many solutions.

If you encounter #NUM!, verify the determinant using the MDETERM function on your coefficient matrix range. If MDETERM returns zero, the system must be re-evaluated as it is mathematically unsolvable by matrix inversion.

A #VALUE! error typically indicates a non-mathematical issue, such as supplying non-numeric data to the matrix arrays or, more commonly, attempting to multiply matrices with incompatible dimensions in the MMULT function. Always ensure that the number of columns in the first array (the output of MINVERSE) exactly matches the number of rows in the second array (the constant matrix B).