Learning Matrix Multiplication: A Step-by-Step Guide (3×3 by 3×2 Matrices)

Authored by

Mohammed looti

November 9, 2025

RECOMMENDED CITATION

Mohammed looti (2025). *Learning Matrix Multiplication: A Step-by-Step Guide (3×3 by 3×2 Matrices)*. PSYCHOLOGICAL STATISTICS. Retrieved from https://statistics.arabpsychology.com/?p=14397

This comprehensive tutorial is designed to guide you step-by-step through the process of performing matrix multiplication when dealing with a 3×3 matrix and a 3×2 matrix. Mastering this specific calculation is a crucial step in understanding complex mathematical operations used widely in data science, engineering, and computer graphics.

Fundamentals of Matrix Conformability and Resulting Dimensions

Matrix multiplication is a foundational concept in Linear algebra, governed by a strict rule of compatibility known as conformability. Before multiplication can proceed, the number of columns in the first matrix must exactly match the number of rows in the second matrix. Failure to meet this requirement means the operation is undefined.

In this scenario, we are multiplying a 3×3 matrix (Matrix A) by a 3×2 matrix (Matrix B). Matrix A has 3 columns, and Matrix B has 3 rows. Since 3 equals 3, the matrices are conformable, and the multiplication is entirely valid. This compatibility rule dictates not just feasibility, but also the shape of the resulting product matrix.

The resulting product matrix, C, will inherit the number of rows from the first matrix (3) and the number of columns from the second matrix (2). Therefore, the final product C will have dimensions of 3×2. Understanding this rule is essential for correctly setting up and verifying matrix computations.

Defining the Input Matrices for Calculation

To proceed with the calculation, we first clearly define the structure of the two input matrices. Matrix A is a **3×3** matrix, meaning it possesses three rows and three columns. We can visualize its general form with subscript notation representing the row (i) and column (j) position of each element:

```
table {
border-collapse: collapse;
border-spacing: 0;
padding: 0;
}
td.tdleft {
border-top: solid 1px #000;
border-bottom: solid 1px #000;
border-left: solid 1px #000;
width: 5px;
padding: 0;
}
```

```
td.tdreg {
padding: 2px 1px;
text-align: center;
border-bottom: solid 1px #fff;
}
td.tdright {
border-top: solid 1px #000;
border-bottom: solid 1px #000;
border-right: solid 1px #000;
width: 5px;
padding: 0;
}
.super_short2 {
max-width: 250px;
margin: 5px auto;
color: blue;
}
```

A =

	A11	A12	A13	
A21	A22	A23		
A31	A32	A33		

Next, we define matrix B, which is a **3×2** matrix. This second matrix dictates the column count of the final product and provides the necessary three rows for multiplication with A's three columns.

```
table {
border-collapse: collapse;
border-spacing: 0;
padding: 0;
}
td.tdleft {
border-top: solid 1px #000;
border-bottom: solid 1px #000;
border-left: solid 1px #000;
width: 5px;
padding: 0;
}
td.tdreg {
```

```
padding: 2px 1px;
text-align: center;
border-bottom: solid 1px #fff;
}
td.tdright {
border-top: solid 1px #000;
border-bottom: solid 1px #000;
border-right: solid 1px #000;
width: 5px;
padding: 0;
}
.short2 {
max-width: 250px;
margin: 5px auto;
color: red;
}
```

B =

	B11	B12	
B21	B22		
B31	B32		

Deriving the General Formula for A × B

The product matrix, C = A × B, is calculated using the fundamental matrix multiplication rule: the row-by-column multiplication. To determine the value of any specific element Cij (the element located in row *i* and column *j* of the result), we must compute the dot product of the *i*-th row of matrix A and the *j*-th column of matrix B.

Since the inner dimensions of A and B are both 3, each resulting element Cij will be the sum of three distinct products. For instance, C11 is calculated by multiplying the elements of row 1 in A by the corresponding elements of column 1 in B, and summing them: (A11*B11) + (A12*B21) + (A13*B31).

Applying this fundamental rule across all possible combinations (3 rows in A and 2 columns in B) yields the comprehensive 3×2 product matrix C, as visualized in the formula below:

```
table {
border-collapse: collapse;
```

```
border-spacing: 0;
padding: 0;
}
td.tdleft {
border-top: solid 1px #000;
border-bottom: solid 1px #000;
border-left: solid 1px #000;
width: 5px;
padding: 0;
}
td.tdreg {
padding: 2px 1px;
text-align: center;
border-bottom: solid 1px #fff;
}
td.tdright {
border-top: solid 1px #000;
border-bottom: solid 1px #000;
border-right: solid 1px #000;
width: 5px;
padding: 0;
}
.long{
margin: 5px auto;
color: #000000;
}
.red {
color: red;
}
.blue {
color: blue;
}
```

A x B =

	A11*B11+A12*B21+A13*B31	A11*B12+A12*B22+A13*B32	
A21*B11+A22*B21+A23*B31	A21*B12+A22*B22+A23*B32		
A31*B11+A32*B21+A33*B31	A31*B12+A32*B22+A33*B32		

This detailed expansion visually confirms the structure of the resulting **3×2 matrix**, validating the conformability rule established earlier.

Example 1: Numerical Calculation using C and D

To solidify our grasp of this process, we now apply the rules of matrix multiplication using specific numerical examples. Our first calculation involves a **3×3** matrix C and a **3×2** matrix D, defined as follows:

```
table {
border-collapse: collapse;
border-spacing: 0;
padding: 0;
}
td.tdleft {
border-top: solid 1px #000;
border-bottom: solid 1px #000;
border-left: solid 1px #000;
width: 5px;
padding: 0;
}
td.tdreg {
padding: 2px 1px;
text-align: center;
border-bottom: solid 1px #fff;
}
td.tdright {
border-top: solid 1px #000;
border-bottom: solid 1px #000;
border-right: solid 1px #000;
width: 5px;
padding: 0;
}
.ex3_3 {
max-width: 250px;
margin: 5px auto;
color: #000000;
}
```

C =

	-3	5	4	
1	2	3		
-1	0	2		

Matrix D, the 3×2 component, provides the two columns for the resultant matrix:

```
table {
border-collapse: collapse;
border-spacing: 0;
padding: 0;
}
td.tdleft {
border-top: solid 1px #000;
border-bottom: solid 1px #000;
border-left: solid 1px #000;
width: 5px;
padding: 0;
}
td.tdreg {
padding: 2px 1px;
text-align: center;
border-bottom: solid 1px #fff;
}
td.tdright {
border-top: solid 1px #000;
border-bottom: solid 1px #000;
border-right: solid 1px #000;
width: 5px;
padding: 0;
}
.short {
max-width: 250px;
margin: 5px auto;
color: #000000;
}
```

D =

	2	1	
5	1		
0	-1		

The product C × D requires six separate dot product calculations. The intermediate matrix shown below visualizes the multiplication of the rows of C by the columns of D before the final summation:

```
table {
border-collapse: collapse;
border-spacing: 0;
padding: 0;
}
td.tdleft {
border-top: solid 1px #000;
border-bottom: solid 1px #000;
border-left: solid 1px #000;
width: 5px;
padding: 0;
}
td.tdreg {
padding: 2px 1px;
text-align: center;
border-bottom: solid 1px #fff;
}
td.tdright {
border-top: solid 1px #000;
border-bottom: solid 1px #000;
border-right: solid 1px #000;
width: 5px;
padding: 0;
}
.medium {
max-width: 500px;
margin: 5px auto;
color: #000000;
}
.red {
```

```
color: red;
}
.blue {
color: blue;
}
```

C x D =

	-3*2 + 5*5 + 4*0	-3*1 + 5*1 + 4*-1	
1*2 + 2*5 + 3*0	1*1 + 2*1 + 3*-1		
-1*2 + 0*5 + 2*0	-1*1 + 0*1 + 2*-1		

By performing the necessary arithmetic (e.g., Row 1, Column 1 calculation: -6 + 25 + 0 = 19; Row 1, Column 2 calculation: -3 + 5 - 4 = -2), we arrive at the final 3×2 product matrix:

```
table {
border-collapse: collapse;
border-spacing: 0;
padding: 0;
}
td.tdleft {
border-top: solid 1px #000;
border-bottom: solid 1px #000;
border-left: solid 1px #000;
width: 5px;
padding: 0;
}
td.tdreg {
padding: 2px 1px;
text-align: center;
border-bottom: solid 1px #fff;
}
td.tdright {
border-top: solid 1px #000;
border-bottom: solid 1px #000;
border-right: solid 1px #000;
width: 5px;
padding: 0;
}
.short {
```

```
max-width: 250px;
margin: 5px auto;
color: #000000;
}
```

C x D =

	19	-2	
12	0		
-2	-3		

Example 2: Reinforcing the Method with E and F

Our second illustrative example uses matrices E (**3×3**) and F (**3×2**) to further reinforce the mechanical process of row-by-column multiplication. Consistent practice with these mechanics is vital for developing intuition necessary for solving more complex systems within Linear algebra.

```
table {
border-collapse: collapse;
border-spacing: 0;
padding: 0;
}
td.tdleft {
border-top: solid 1px #000;
border-bottom: solid 1px #000;
border-left: solid 1px #000;
width: 5px;
padding: 0;
}
td.tdreg {
padding: 2px 1px;
text-align: center;
border-bottom: solid 1px #fff;
}
td.tdright {
border-top: solid 1px #000;
border-bottom: solid 1px #000;
border-right: solid 1px #000;
width: 5px;
padding: 0;
```

```
}
.ex3_3 {
max-width: 250px;
margin: 5px auto;
color: #000000;
}
```

E =

	2	8	1	
3	3	0		
0	1	2		

Matrix F is the **3×2** component that defines the structure of the resulting product:

```
table {
border-collapse: collapse;
border-spacing: 0;
padding: 0;
}
td.tdleft {
border-top: solid 1px #000;
border-bottom: solid 1px #000;
border-left: solid 1px #000;
width: 5px;
padding: 0;
}
td.tdreg {
padding: 2px 1px;
text-align: center;
border-bottom: solid 1px #fff;
}
td.tdright {
border-top: solid 1px #000;
border-bottom: solid 1px #000;
border-right: solid 1px #000;
width: 5px;
padding: 0;
}
.short {
```

max-width: 250px;
margin: 5px auto;
color: #000000;
}

F =	<table><tr><td></td><td>-2</td><td>-2</td><td></td></tr><tr><td>3</td><td>1</td><td></td><td></td></tr><tr><td>4</td><td>10</td><td></td><td></td></tr></table>

The intermediate step for E × F details the six individual dot product calculations before final summation. Notice how the calculation ensures that the number of terms added in each cell corresponds to the inner dimension (3):

table {
border-collapse: collapse;
border-spacing: 0;
padding: 0;
}
td.tdleft {
border-top: solid 1px #000;
border-bottom: solid 1px #000;
border-left: solid 1px #000;
width: 5px;
padding: 0;
}
td.tdreg {
padding: 2px 1px;
text-align: center;
border-bottom: solid 1px #fff;
}
td.tdright {
border-top: solid 1px #000;
border-bottom: solid 1px #000;
border-right: solid 1px #000;
width: 5px;
padding: 0;
}
.medium {

max-width: 500px;
margin: 5px auto;
color: #000000;
}
.red {
color: red;
}
.blue {
color: blue;
}

E x F =

	2*-2 + 8*3 + 1*4	2*-2 + 8*1 + 1*10	
3*-2 + 3*3 + 0*4	3*-2 + 3*1 + 0*10		
0*-2 + 1*3 + 2*4	0*-2 + 1*1 + 2*10		

Simplifying these terms (e.g., E × F (1,1) = -4 + 24 + 4 = 24) provides the final 3×2 product matrix:

table {
border-collapse: collapse;
border-spacing: 0;
padding: 0;
}
td.tdleft {
border-top: solid 1px #000;
border-bottom: solid 1px #000;
border-left: solid 1px #000;
width: 5px;
padding: 0;
}
td.tdreg {
padding: 2px 1px;
text-align: center;
border-bottom: solid 1px #fff;
}
td.tdright {
border-top: solid 1px #000;
border-bottom: solid 1px #000;
border-right: solid 1px #000;

width: 5px;
padding: 0;
}
.short {
max-width: 250px;
margin: 5px auto;
color: #000000;
}

E x F =		24	14	
	3	-3		
	11	21		

Example 3: Handling Zeros and Negatives with G and H

Our final example utilizes matrix G (**3×3**) and matrix H (**3×2**). This set specifically highlights how the presence of zeros and negative numbers impacts the intermediate products, without changing the core mathematical rules of matrix multiplication. Matrix G is defined below:

table {
border-collapse: collapse;
border-spacing: 0;
padding: 0;
}
td.tdleft {
border-top: solid 1px #000;
border-bottom: solid 1px #000;
border-left: solid 1px #000;
width: 5px;
padding: 0;
}
td.tdreg {
padding: 2px 1px;
text-align: center;
border-bottom: solid 1px #fff;
}
td.tdright {
border-top: solid 1px #000;

border-bottom: solid 1px #000;
border-right: solid 1px #000;
width: 5px;
padding: 0;
}
.ex3_3 {
max-width: 250px;
margin: 5px auto;
color: #000000;
}

G =		-1	0	0	
	7	1	0		
	2	4	6		

Matrix H, defined as a **3×2** matrix, completes the setup for this multiplication:

table {
border-collapse: collapse;
border-spacing: 0;
padding: 0;
}
td.tdleft {
border-top: solid 1px #000;
border-bottom: solid 1px #000;
border-left: solid 1px #000;
width: 5px;
padding: 0;
}
td.tdreg {
padding: 2px 1px;
text-align: center;
border-bottom: solid 1px #fff;
}
td.tdright {
border-top: solid 1px #000;
border-bottom: solid 1px #000;
border-right: solid 1px #000;

width: 5px;
padding: 0;
}
.short {
max-width: 250px;
margin: 5px auto;
color: #000000;
}

H =				
		4	5	
	9	2		
	0	1		

We calculate the resulting product G × H by applying the row-by-column multiplication. Note how the zeros in the first row of G simplify the calculations for the resulting elements G×H(1,1) and G×H(1,2):

table {
border-collapse: collapse;
border-spacing: 0;
padding: 0;
}
td.tdleft {
border-top: solid 1px #000;
border-bottom: solid 1px #000;
border-left: solid 1px #000;
width: 5px;
padding: 0;
}
td.tdreg {
padding: 2px 1px;
text-align: center;
border-bottom: solid 1px #fff;
}
td.tdright {
border-top: solid 1px #000;
border-bottom: solid 1px #000;
border-right: solid 1px #000;

```
width: 5px;
padding: 0;
}
.medium {
max-width: 500px;
margin: 5px auto;
color: #000000;
}
.red {
color: red;
}
.blue {
color: blue;
}
```

G x H =

	-1*4 + 0*9 + 0*0	-1*5 + 0*2 + 0*1	
7*4 + 1*9 + 0*0	7*5 + 1*2 + 0*1		
2*4 + 4*9 + 6*0	2*5 + 4*2 + 6*1		

After summing the products in each position (e.g., G × H (3,1) = 8 + 36 + 0 = 44), we derive the final resulting 3×2 matrix:

```
table {
border-collapse: collapse;
border-spacing: 0;
padding: 0;
}
td.tdleft {
border-top: solid 1px #000;
border-bottom: solid 1px #000;
border-left: solid 1px #000;
width: 5px;
padding: 0;
}
td.tdreg {
padding: 2px 1px;
text-align: center;
border-bottom: solid 1px #fff;
```

```
}
td.tdright {
border-top: solid 1px #000;
border-bottom: solid 1px #000;
border-right: solid 1px #000;
width: 5px;
padding: 0;
}
.short {
max-width: 250px;
margin: 5px auto;
color: #000000;
}
```

G x H =

	-4	-5	
37	37		
44	24		

Summary and Further Learning

The process of multiplying a 3×3 matrix by a 3×2 matrix relies entirely on the principle of row-by-column dot product summation. As demonstrated through the detailed examples, the crucial first step is verifying conformability (3 columns in A matching 3 rows in B), which guarantees a resulting 3×2 matrix.

Mastering matrix arithmetic, especially involving non-square dimensions, is foundational for advanced mathematical applications, including solving systems of linear equations and transformations in computational geometry. If you are looking to explore other matrix operations beyond the 3×3 by 3×2 case, consider reviewing tutorials on commutative properties or inverse matrix calculations.