

In this brief blog post I post a small code that can be used to solve a system of linear equations using SciPy package of Python and in particular the linalg module within the SciPy package.

Documentation: <https://docs.scipy.org/doc/scipy-0.14.0/reference/generated/scipy.linalg.solve.html>

The code is pretty much self-explanatory, and there is also a YouTube video where I walk you through the procedure.

CODE:

```
import numpy as np
from scipy import linalg
#Solve a system of equations A.X=B

#Define the LHS coefficient matrix A
A = np.array([ [1, 3, -2], [3, 5, 6], [2, 4, 3] ])
#Print the matrix A
print(A)
#Define the RHS column vector B
B = np.array([ [5, 7, 8] ]) #This is a single row
#Transpose it to make a column
B = B.T
#Print the RHS vector
print(B)
#Solve the system of equations and store the result in X
X = linalg.solve(A, B)
#Print the solution
print(X)
```

YouTube Tutorial



Manas Sharma

I'm a physicist specializing in computational material science with a PhD in Physics from Friedrich-Schiller University Jena, Germany. I write efficient codes for simulating light-matter interactions at atomic scales. I like to develop Physics, DFT, and Machine Learning related apps and software from time to time. Can code in most of the popular languages. I like to share my knowledge in Physics and applications using this Blog and a YouTube channel.

manas.bragitoff.com/









Share this:

Click to share on Facebook (Opens in new window)

Click to share on Twitter (Opens in new window)

Click to share on WhatsApp (Opens in new window)

Click to share on Pinterest (Opens in new window)

Click to share on Reddit (Opens in new window)

Click to share on LinkedIn (Opens in new window)

Click to email a link to a friend (Opens in new window)

Click to print (Opens in new window)

Click to share on Tumblr (Opens in new window)

Click to share on Pocket (Opens in new window)

Click to share on Telegram (Opens in new window)

[wpdon id="7041" align="center"]