ACSE_Ia

Jakob Torben

Oct 30, 2020

CONTENTS

`acse_la` is a linear algebra library that implements Gaussian Elimination, matrix multiplication and calculating the determinant.

CHAPTER

ONE

A GAUSSIAN ELIMINATION ROUTINE

This package implements Gaussian elimination [1] for list of lists and `numpy.ndarray` objects, along with hand-written matrix multiplication.

See `acse_la.gauss()` and `acse_la.gauss.matmul()` for more information.

`acse_la.gauss.`**`gauss`**(*a*, *b*)

Given two matrices, *a* and *b*, with *a* square, the determinant of *a* and a matrix *x* such that a*x = b are returned. If *b* is the identity, then *x* is the inverse of *a*.

Parameters

- **a** (*np.array or list of lists*) – 'n x n' array
- **b** (*np. array or list of lists*) – 'm x n' array

Examples

```
>>> a = [[2, 0, -1], [0, 5, 6], [0, -1, 1]]
>>> b = [[2], [1], [2]]
>>> det, x = gauss(a, b)
>>> det
22.0
>>> x
[[1.5], [-1.0], [1.0]]
>>> A = [[1, 0, -1], [-2, 3, 0], [1, -3, 2]]
>>> I = [[1, 0, 0], [0, 1, 0], [0, 0, 1]]
>>> Det, Ainv = gauss(A, I)
>>> Det
3.0
>>> Ainv
[[2.0, 1.0, 1.0],
[1.3333333333333333, 1.0, 0.6666666666666666],
[1.0, 1.0, 1.0]]
```

Notes

See https://en.wikipedia.org/wiki/Gaussian_elimination for further details.

acse_la.gauss.**matmul**(*a*, *b*)

Matrix product of matrix a and matrix b.

Parameters

- **a** (*np.array or list of lists*) – 'n x m' array
- **b** (*np. array or list of lists*) – 'm x l' array

Returns **out** – The matrix product of the inputs.

Return type list of lists

Raises **ValueError** – If the number of columns of *a* is not the same as the number of rows *b*.

Notes

The output dimension depends on the dimensions of the input.

- For input matrices a = n x m and b = m x l, the output matrix will have dimensions n x l.

Examples

For 2-D arrays:

```
>>> a = np.array([[1, 2],
...               [3, 4]])
>>> b = np.array([[5, 1],
...               [6, 2]])
>>> matmul(a, b)
[[17, 5], [39, 11]]
```

For a 2-D array and 1-D array: >>> a = np.array([[1, 2], ... [3, 4]]) >>> b = np.array([5, 6]) >>> matmul(a, b) [17, 39]

acse_la.gauss.**zeromat**(*a*, *b*)

Returns an array with dimension shape, filled with zeros.

Parameters **shape** (*tuple*) – Shape of output matrix

Returns **out** – Matrix with dimension p x q

Return type list of lists

Examples

```
>>> zeromat(2, 3)
[[0, 0, 0], [0, 0, 0]]
>>> zeromat(1, 3)
[[0, 0, 0]]
>>> zeromat(2, 1)
[[0], [0]]
```

CHAPTER

TWO

A DETERMINANT ROUTINE

acse_la.det.**det**(*A*)

Compute the determinant of a matrix.

Parameters **A** (*np.array or list of lists*) – 'N x N' matrix

Returns **out** – The determinant of the matrix.

Return type float

Raises **ValueError** – If input is not square matrix

Notes

- The determinant is computed with LU decomposition, using Crout's method. For further details see: Propp, J. G., Wilson, D. B. 'Numerical Recipes', Cambridge University Press. (1996).
- Pivoting is not yet implemented. Matrices with zero entries along diagonal can have unstable behaviour.

Examples

```
>>> det([[2, 9, 4], [7, 5, 3], [6, 1, 8]])
-360.0
>>> det([[0.5, 1.5], [4.2, 3.9]])
-4.35
```

CHAPTER

THREE

CITATION

[1] https://mathworld.wolfram.com/GaussianElimination.html

[2] Propp, J. G., Wilson, D. B. 'Numerical Recipes', Cambridge University Press. (1996)