MPM_la

unknown

Oct 29, 2021

CONTENTS

CHAPTER

ONE

A GAUSSIAN ELIMINATION ROUTINE

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

See *mpm_la.gauss()* and `mpm_la.gauss.matmul()` for more information.

`mpm_la.`**`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

```
>>> from mpm_la import gauss
>>> 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]]
>>> from mpm_la import gauss
>>> 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
```

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

Notes

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

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

Given two matrices, *a* and *b*. First, determine the shape of the result matrix after multiplication according to the shapes of the matrices a and b, and generate a zero matrix of the corresponding shape. Next, complete the matrix multiplication and return the result matrix.

Parameters

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

Examples

```
>>> from mpm_la import matmul
>>> a=[[1,2],[3,4]]
>>> b=[[5],[6]]
>>> res_mul=matmul(a,b)
>>> res_mul
[[17], [39]]
```

```
>>> from mpm_la import matmul
>>> a=[[1,2],[3,4]]
>>> b=[[5,1],[6,2]]
>>> mul=matmul(a,b)
>>> mul
[[17, 5], [39, 11]]
```

mpm_la.gauss.**zeromat**(*p*, *q*)

Given two integers, *p* and *q*. *p* is the number of rows in the first matrix, and *q* is the number of columns in the second matrix. The function will return a matrix with all zero values. The shape of the returned matrix is the same as the shape as a result of multiplying two matrices.

Parameters

- **p** (*Integer*) –
- **q** (*Integer*) –

Examples

```
>>> from mpm_la import zeromat
>>> p = 3
>>> q = 1
>>> res_zero = zeromat(p, q)
>>> res_zero
[[0], [0], [0]]
```

```
>>> from mpm_la import zeromat
>>> p = 4
>>> q = 4
>>> res_zero = zeromat(p, q)
>>> res_zero
[[0, 0, 0, 0], [0, 0, 0, 0], [0, 0, 0, 0], [0, 0, 0, 0]]
```

CHAPTER

TWO

ANOTHER ALGORITHM TO COMPUTE THE DETERMINANT

This package also implements another algorithm for `numpy.ndarray` objects, to compute the determinant of a single square matrix.

See `mpm_la.det()` for more information.

`mpm_la.det.det`(*mat*)

Given one matrix, *mat*, the determinant of *mat* will be returned.

Parameters **`mat`** (*`np.array or list of lists`*) – 'n x n' array

References

PYTHON MODULE INDEX

m