

# 矩阵计算库 API

高飞

2019 年 1 月 24 日

目录

## 1 简介

- 包含矩阵的常用操作，如四则运算、求逆矩阵、求特征值、解线性方程组等。
- C++ 编译器需支持 C++11 及以上标准。

## 2 矩阵构造

### 2.1 构造函数

```
Matrix();
Matrix(const usize& m, const usize& n);
Matrix(const std::initializer_list<std::initializer_list<double>>& m);
Matrix(const Matrix& other);
Matrix(Matrix&& other);
```

### 2.2 Examples

```
Matrix A;
Matrix B(2,3);
Matrix C1({{1,2,3}});
Matrix C2({{1},{2},{3}});
Matrix D({1,2,3},{4,5,6});
Matrix E(D);
```

**Result:**

$$A = (), \quad B = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \quad C1 = \begin{pmatrix} 1 & 2 & 3 \end{pmatrix}, \quad C2 = \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix}, \quad D = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{pmatrix}, \quad E = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{pmatrix}$$

## 3 常用矩阵

- eye
- ones
- zeros
- hilb
- rand
- randn
- diag
- subMat

## 4 元素索引

支持 `[]` 操作符和 `at()` 函数进行矩阵元素索引。

**Example:**

```
Matrix M({1,2,3},{4,5,6});  
M[1][1] = 0;           // [] 操作符  
M.at(0,2) = 0.5;       // at() 函数
```

**Result:**  $M = \begin{pmatrix} 1 & 2 & 0.5 \\ 4 & 0 & 6 \end{pmatrix}$

`[]` 操作符与 `at()` 函数的区别: `[]` 操作符不进行下标越界检查, `at()` 函数会进行下标检查, 若下标越界则抛出 `std::out_of_range` 异常。

## 5 四则运算

- 赋值运算
- 加法
- 减法
- 乘法
- 除法

## 6 矩阵基本操作

## 7 矩阵分解

- QR 分解
- LU 分解 (Doolittle 分解)
- SVD 分解