

成像算法基础

插值

插值拟合回归最优化

1 插值和拟合

拟合:是一种把现有数据透过数学方法来代入一条数式的表示方式

插值:(英语: interpolation) 是一种通过已知的、离散的数据点, 在范围内推求新数据点的过程或方法.

2最优化

最优化方法, 是指解决最优化问题的方法。所谓最优化问题, 指在某些约束条件下, 决定某些可选择的变量应该取何值, 使所选定的目标函数达到最优的问题。即运用最新科技手段和处理方法, 使系统达到总体最优.

3 回归

回归分析 (regression analysis)指的是确定两种或两种以上变量间相互依赖的定量关系的一种统计分析方法。

线性回归

非线性回归

4 最小二乘法

最小二乘法Least Square Method, 做为分类回归算法的基础, 有着悠久的历史(由马里·勒让德于1806年提出)。它通过最小化误差的平方和寻找数据的最佳函数匹配。利用最小二乘法可以简便地求得未知的数据, 并使得这些求得的数据与实际数据之间误差的平方和为最小。最小二乘法还可用于曲线拟合。



Scipy



Scipy: 一个用于数学、科学、工程领域的常用软件包.

[scipy.constants](#) 物理和数学常数

[scipy.fftpack](#) 傅里叶变换

[scipy.integrate](#) 集成例程

[scipy.interpolate](#) 插值

[scipy.io](#) 数据输入和输出

[scipy.linalg](#) 线性代数例程

[scipy.optimize](#) 优化

[scipy.signal](#) 信号处理

[scipy.sparse](#) 稀疏矩阵

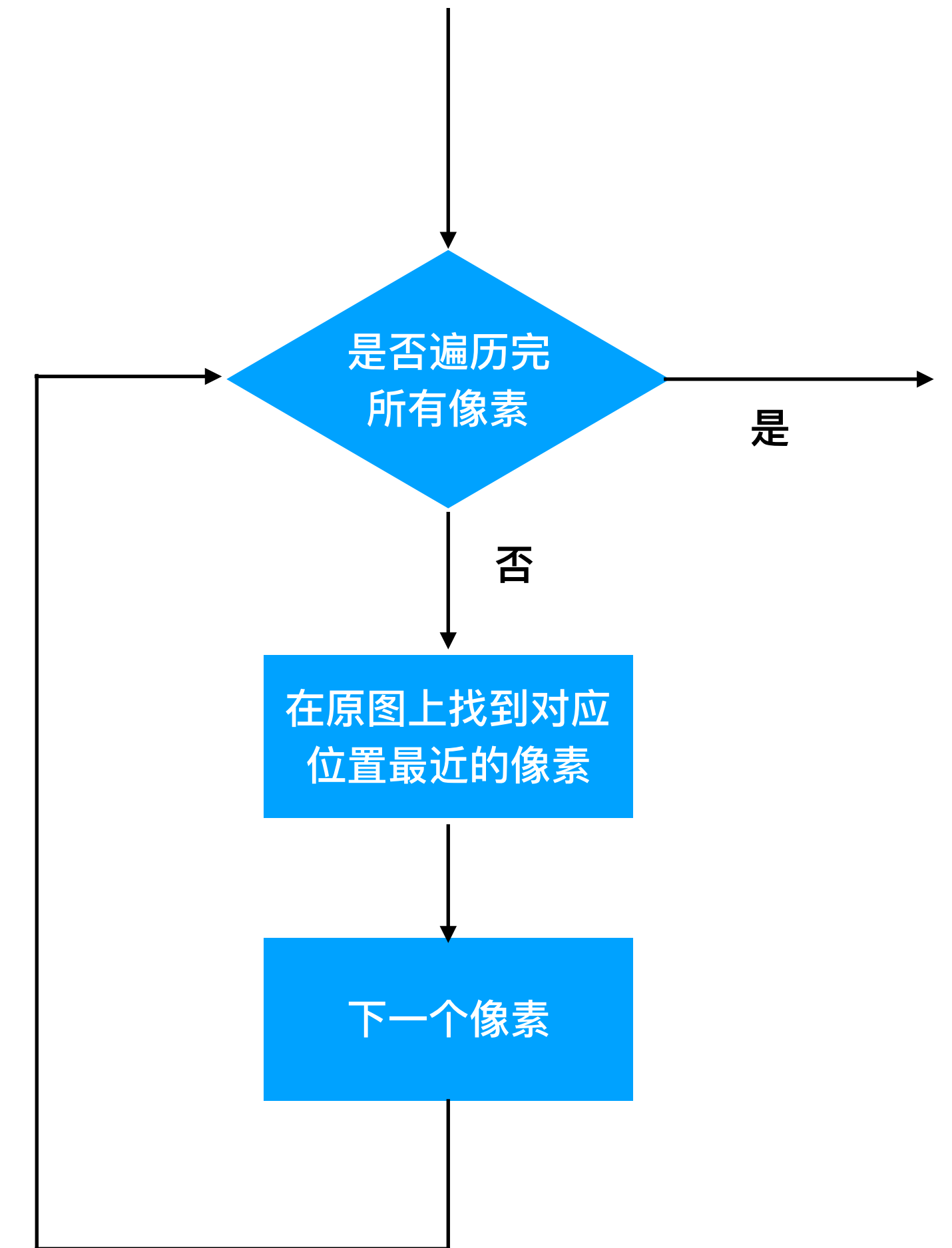
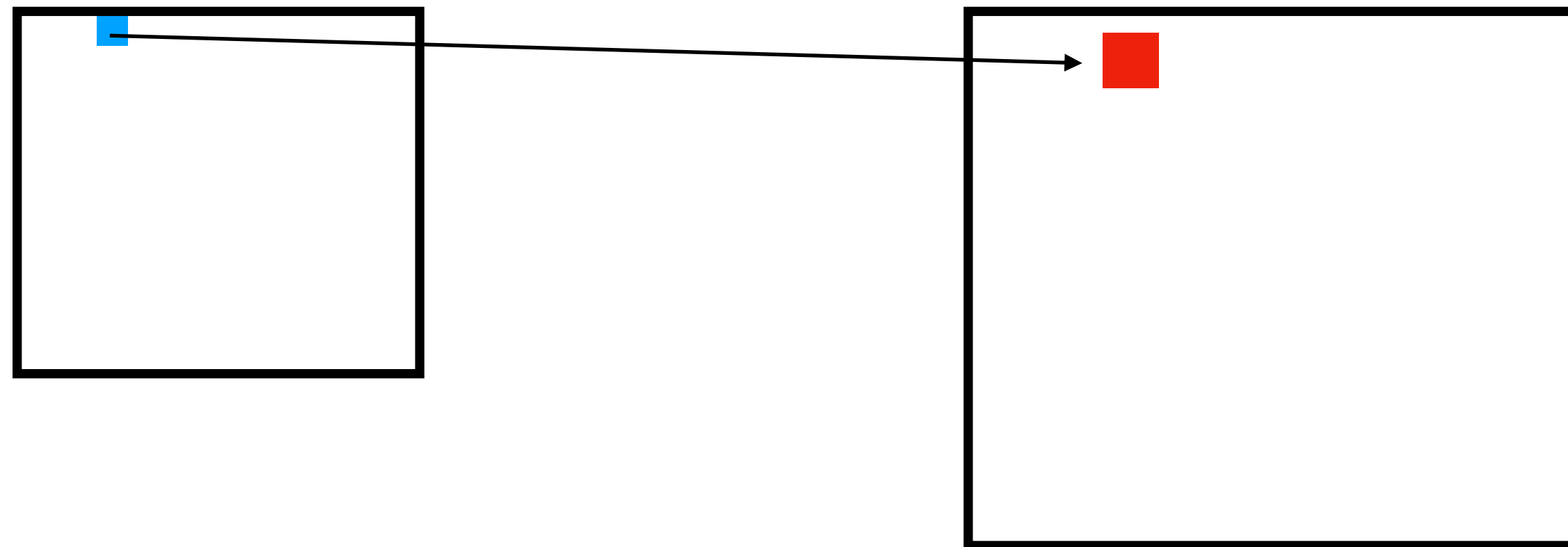
[scipy.spatial](#) 空间数据结构和算法

[scipy.special](#) 任何特殊的数学函数

[scipy.stats](#) 统计

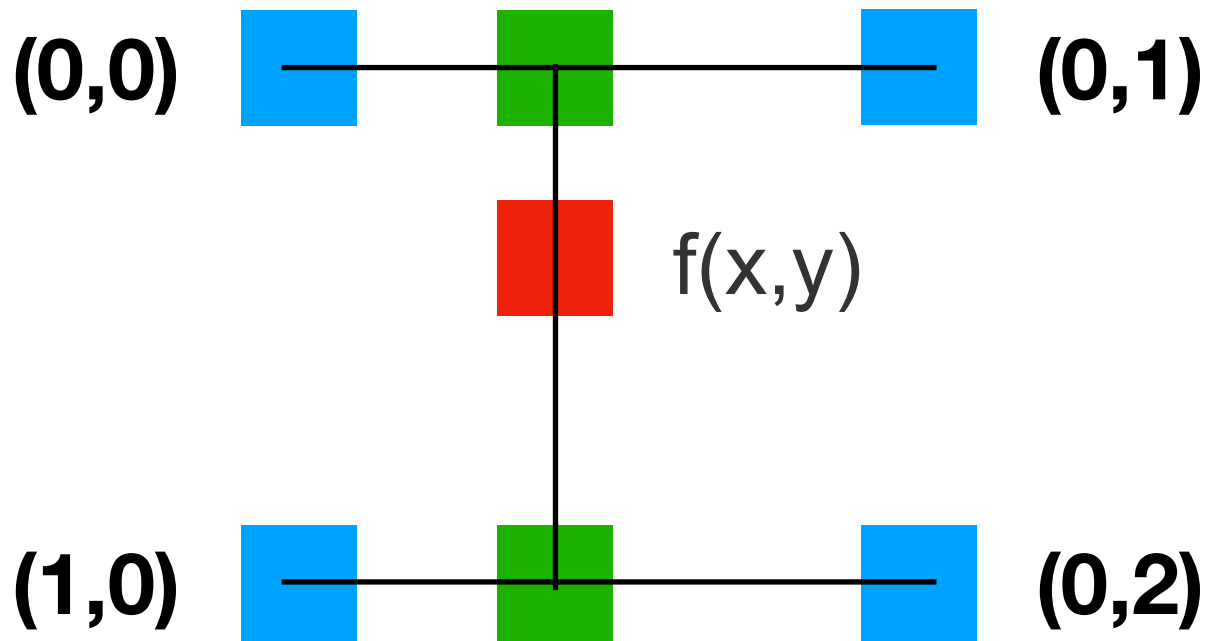
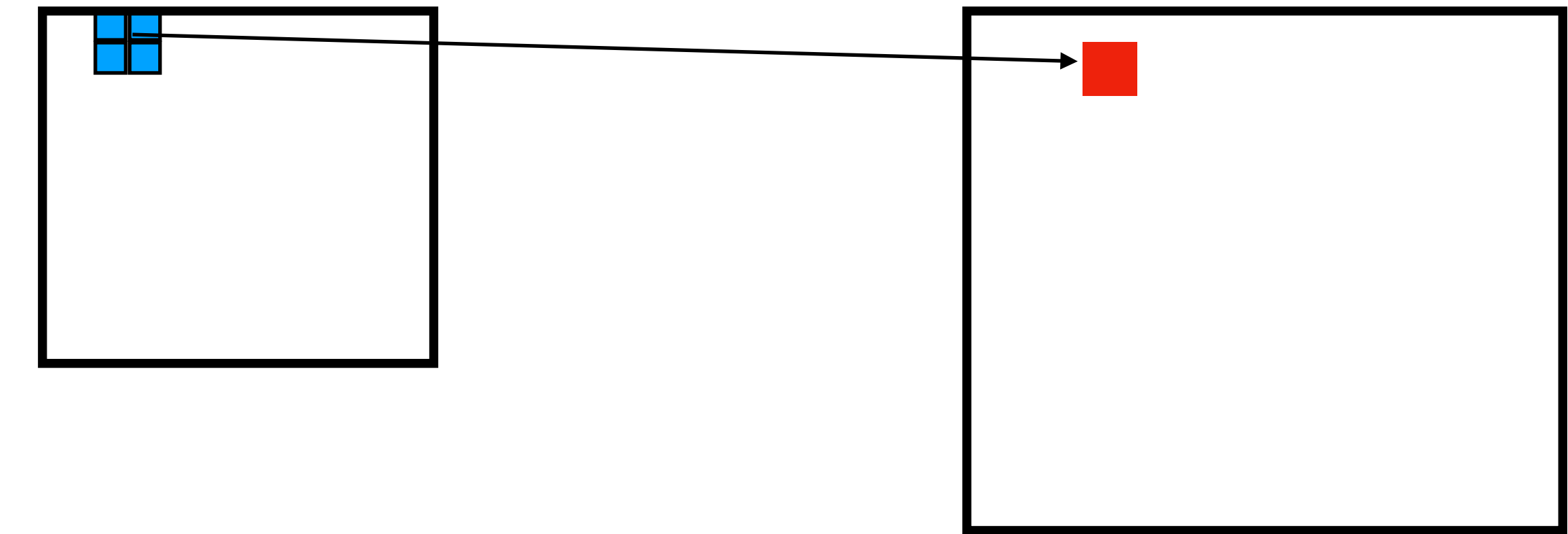
临近插值

为新图的每个像素在原图上找到最近的像素

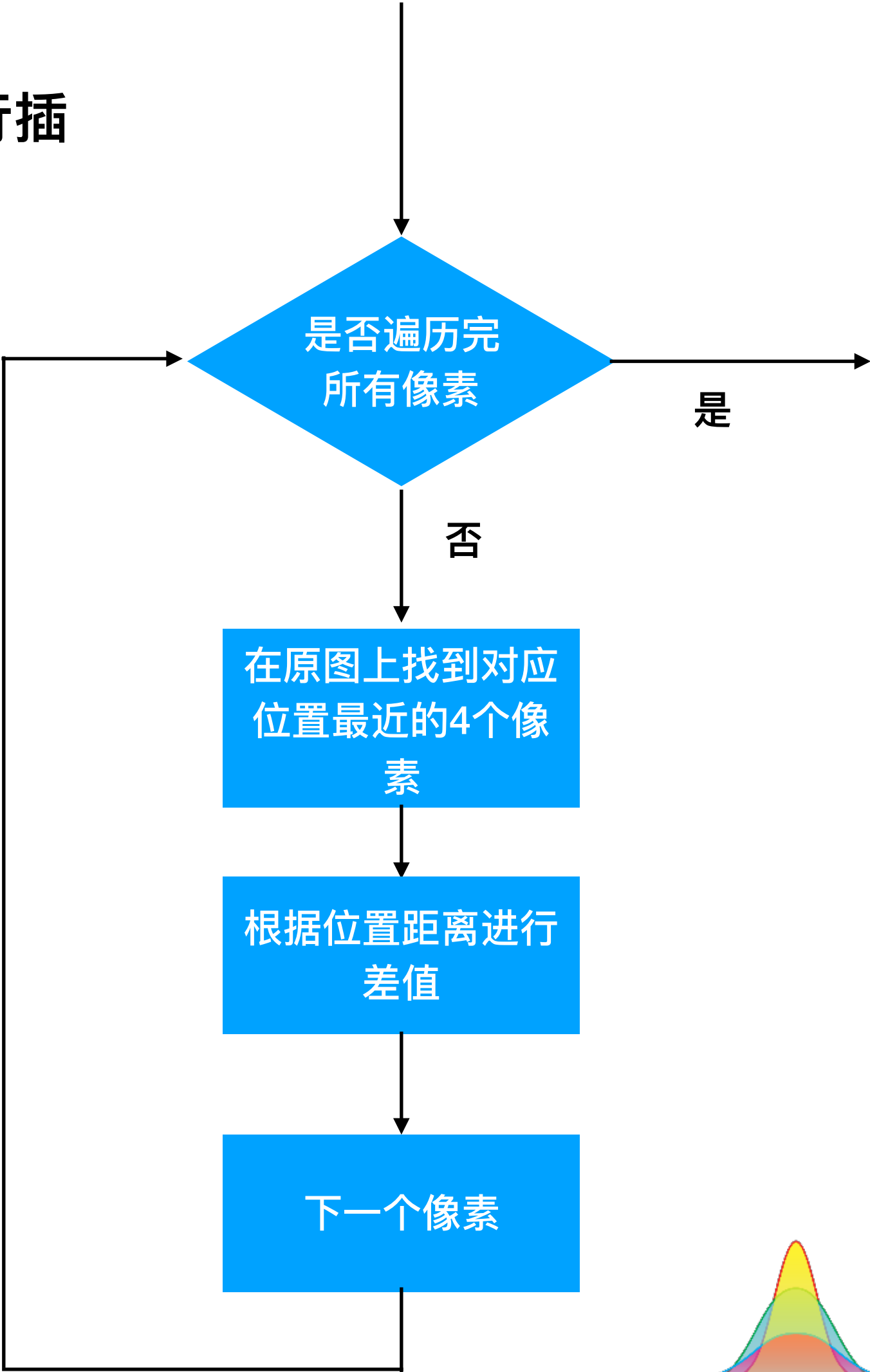


双线性插值

为新图的每个像素在原图上找到最近的4个像素,先两两根据X方向的距离作为权重进行插值,然后再根据Y方向的距离作为权重进行插值

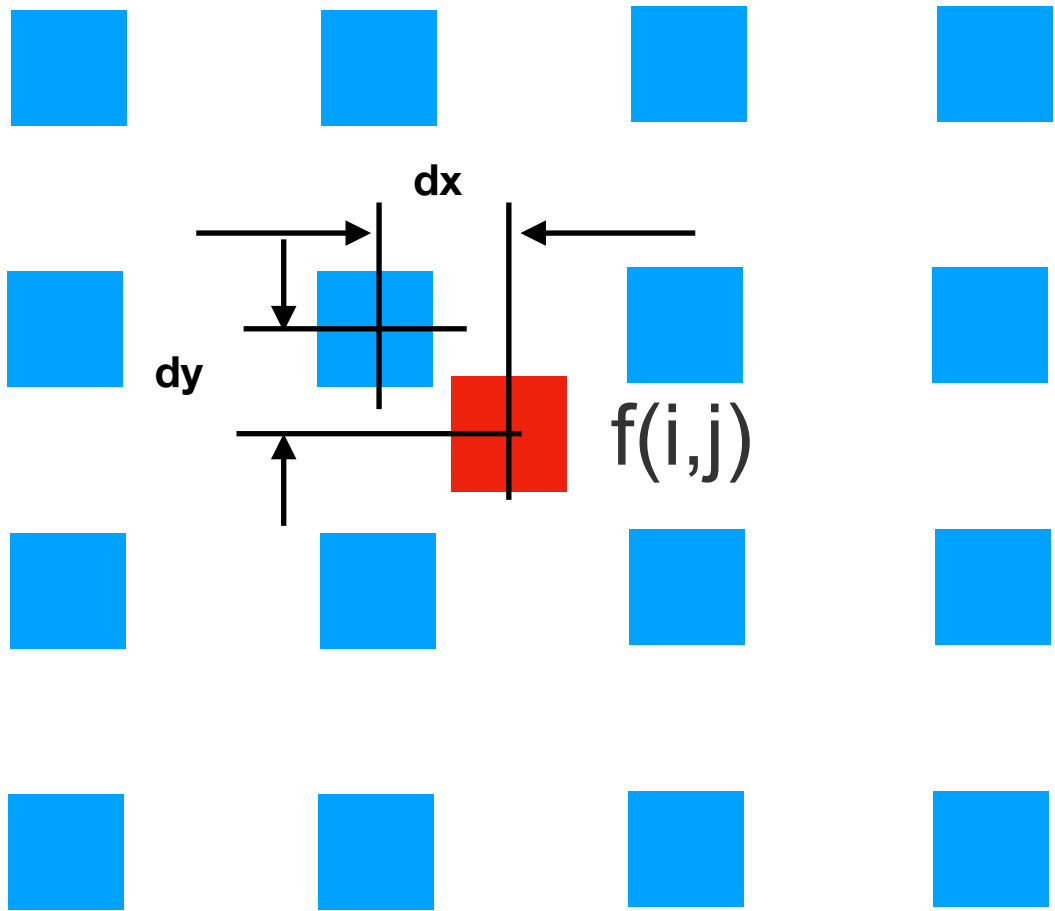
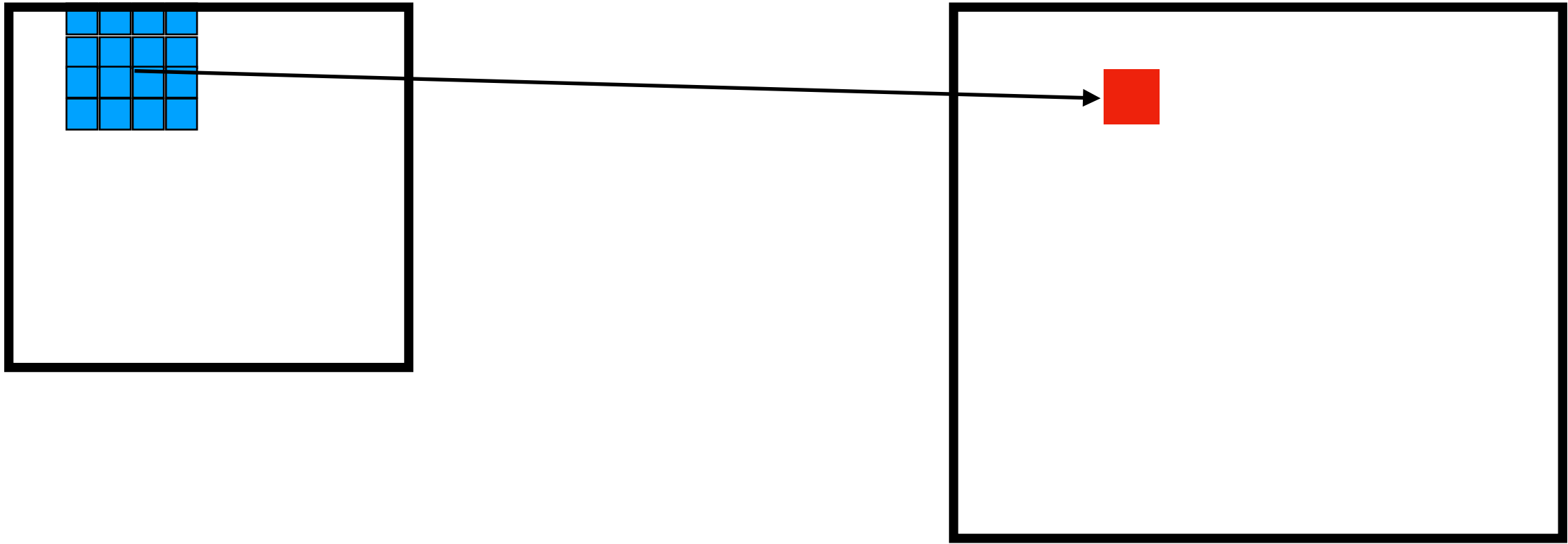


$$f(x,y)=f(0,0)(1-x)(1-y)+f(1,0)x(1-y)+f(0,1)(1-x)y+f(1,1)xy$$

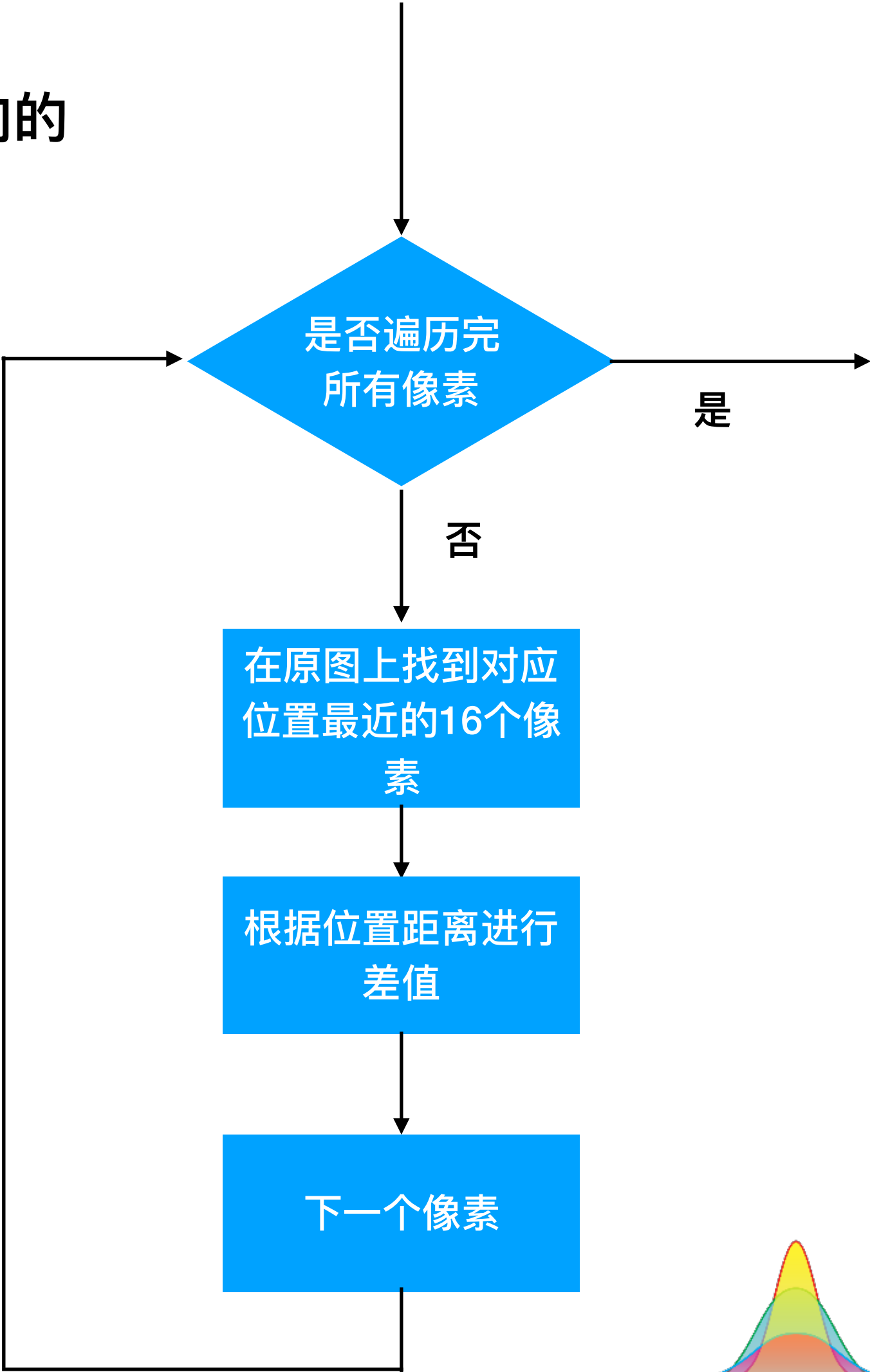


双三次插值

为新图的每个像素在原图上找到最近的16个像素,16个像素分别根据距离像素两个方向的距离进行



$$f(i,j) = \sum_{m=-1}^2 \sum_{n=-1}^2 F(i+m, j+n) * R(m-dx) * R(dy-n)$$



双三次插值的权重函数

三角形

$$R(x) = \begin{cases} x+1 & -1 \leq x \leq 0 \\ 1-x & 0 < x \leq 1 \end{cases}$$



三角取值

Bell钟型

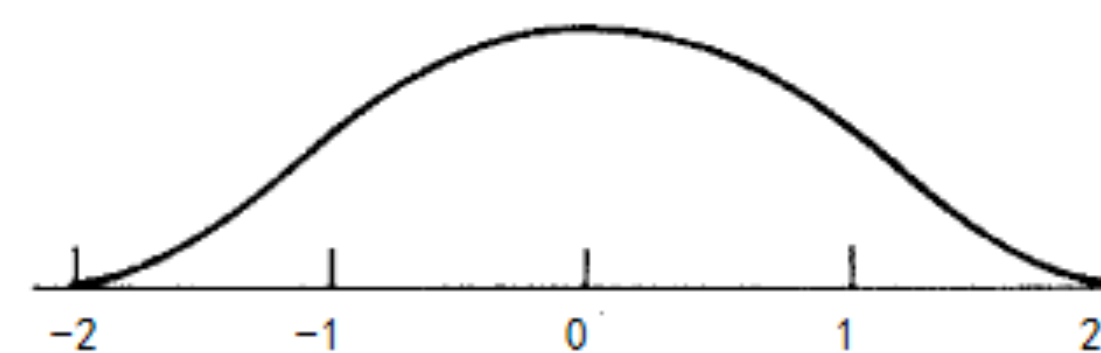
$$R(x) = \begin{cases} \frac{1}{2}(x+\frac{3}{2})^2 & -\frac{3}{2} \leq x \leq -\frac{1}{2} \\ \frac{3}{4} - (x)^2 & -\frac{1}{2} < x \leq \frac{1}{2} \\ \frac{1}{2}(x-\frac{3}{2})^2 & \frac{1}{2} < x \leq \frac{3}{2} \end{cases}$$



钟型分布

Spline条样

$$R(x) = \begin{cases} \frac{2}{3} + \frac{1}{2}|x|^3 - (x)^2 & 0 \leq |x| \leq 1 \\ \frac{1}{6}(2-|x|)^3 & 1 \leq |x| \leq 2 \end{cases}$$



基于B样条曲线卷积插值

系统函数

一维插值

`scipy.interpolate.interp1d`

二维差值

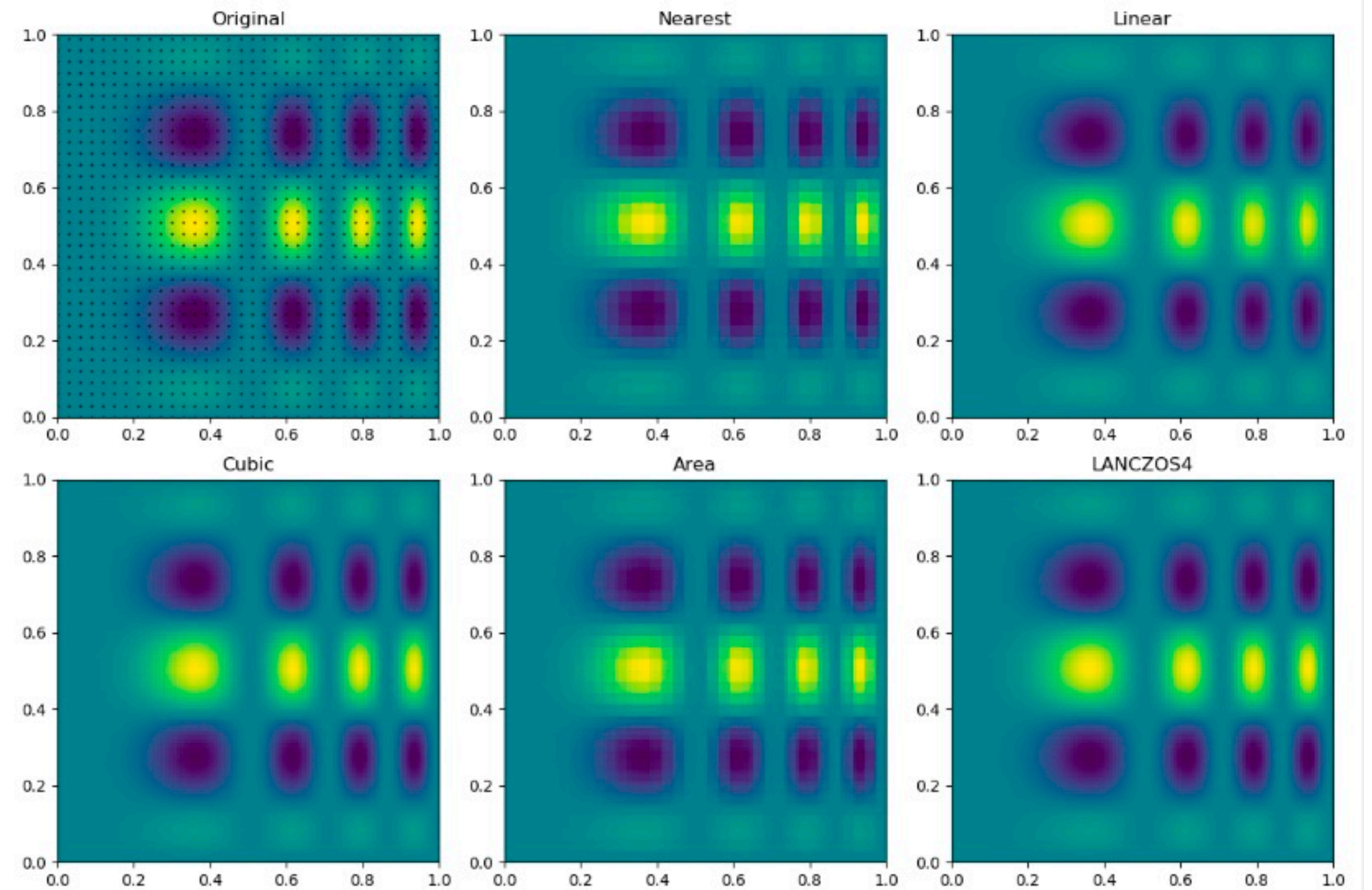
`cv2.resize`

`scipy.interpolate.interp2d`

`scipy.interpolate.grid`

`scipy.interpolate.Rbf`

.....



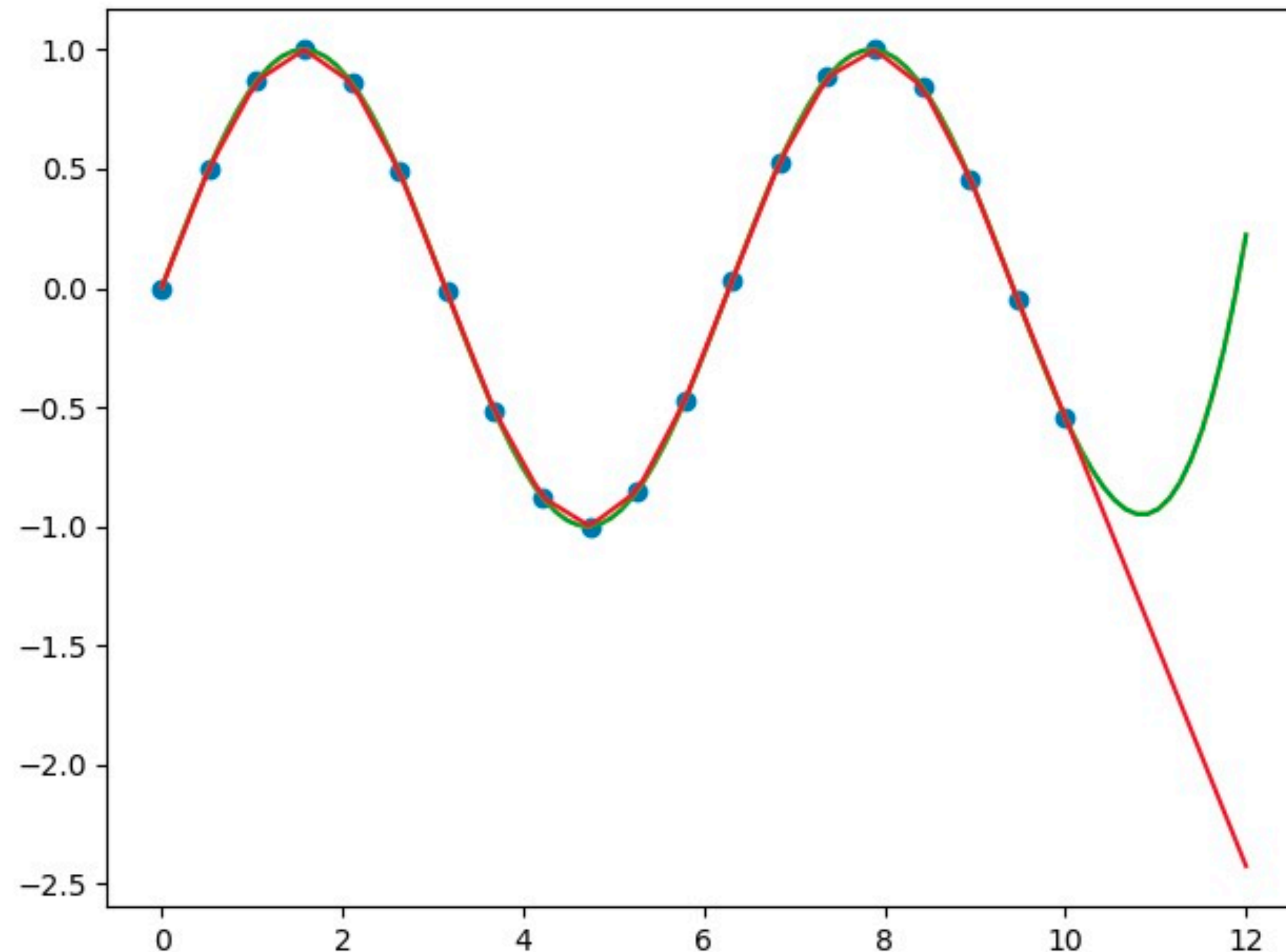
外插这件事情

- 外插(推)是预测

`interpolate.interp1d(x1,y1,kind="cubic",fill_value='extrapolate')`

`interpolate.UnivariateSpline(x1,y1,s=0)`

.



二维扩展

原矩阵

a= [[0 1 2]
[3 4 5]]

reflect

d= [[-8 -7 -6 -5 -4 -3 -2]
[-5 -4 -3 -2 -1 0 1]
[-2 -1 0 1 2 3 4]
[1 2 3 4 5 6 7]
[4 5 6 7 8 9 10]
[7 8 9 10 11 12 13]]

edage

e= [[0 0 0 1 2 2 2]
[0 0 0 1 2 2 2]
[0 0 0 1 2 2 2]
[3 3 3 4 5 5 5]
[3 3 3 4 5 5 5]
[3 3 3 4 5 5 5]]



最小二乘法

- 若A是秩为N的m X n的矩阵,则正规方程组

- $X^T X A = X^T Y$

- 则有唯一解,x为方程组AX=Y的最小二乘解:

- $a = (X^T X)^{-1} X^T Y$

