

## ■ Fourier Transform

- 영상 품질 : signaltonoise

```
def signaltonoise(a, axis=0, ddof=0):
    a = np.asarray(a)
    m = a.mean(axis)
    sd = a.std(axis=axis, ddof=ddof)
    return np.where(sd == 0, 0, m/sd)
```

- 푸리에 변환 이미지 데이터 구조: numpy
- FFT(Fast Fourier Transform)
  - scipy.fftpack : fft2, ifft2().real
  - numpy.fft : fft2, ifft2().real
- numpy.allclose 함수: 모든 요소의 차이가 허용 오차(tolerance) 이내에 있다면 True, 그렇지 않다면 False를 반환

## ■ Convolution

- 컨볼루션: 입력 영상을 위한 필터(마스크(Mask), 커널(Kernel))
- scipy : convolve2d() 함수
  - mode: 출력 영상 크기('full', 'same', 'valid')
  - boundary: 경계 처리 방법('fill', 'wrap', 'symm')

## ■ Spyder 사용

- #%%: Cell 정의

## ■ matplotlib.pyplot.plot

- <https://matplotlib.org>

```
pylab.plot([1, 2, 3, 4], [1, 4, 9, 16], 'ro')
pylab.show()

pylab.plot(["Seoul", "Paris", "Seattle"], [30, 25, 55])
pylab.xlabel('City')
pylab.ylabel('Response')
pylab.title('Experiment Result')
pylab.show()

y = [5, 3, 7, 10, 9, 5, 3.5, 8]
x = range(len(y))
pylab.bar(x, y, width=0.7, color="blue")
pylab.show()
```