UDFT Documentation

Release 3.6.0

François Orieux

Aug 30, 2022

CONTENTS

This module implements unitary discrete Fourier transform, that is orthonormal. This module existed before the introduction of the `norm="ortho"` keyword and is now a very (very) thin wrapper around Numpy or pyFFTW (maybe others in the future), mainly done for my *personal usage*. There is also functions related to Fourier and convolution like `ir2fr`.

It is useful for convolution [1]: they respect the Perceval equality $\|x\|_2^2 = \|X\|_2^2$, e.g., the value of the null frequency X_0 is equal to

$$X_0 = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} x_n, \text{ and } x_0 = \frac{1}{\sqrt{N}} \sum_{n'=0}^{N-1} X_{n'}.$$

```
[1] B. R. Hunt "A matrix theory proof of the discrete convolution theorem",
IEEE Trans. on Audio and Electroacoustics, vol. au-19, no. 4, pp. 285-288,
dec. 1971
```

If you are having issues, please let me know

francois.orieux AT l2s.centralesupelec.fr

CHAPTER

ONE

INSTALLATION AND DOCUMENTATION

UDFT is just the file `udft.py` and depends on `numpy` and Python 3.7 only. I recommend using poetry for installation

```
poetry add udft
```

For potential better performance, pyFFTW is optional and installable with

```
poetry add udft[fftw]
```

The package is available with pip also. For a quick and dirty installation, just copy the `udft.py` file: it is quite stable, follow the Semantic Versioning, and major changes are unlikely.

CHAPTER

TWO

LICENSE

The code is in the public domain.

2.1 The `LIB` module variable

The `LIB` variable specifies the default library to use by the module. The Numpy library is used by default with `LIB` sets to `"numpy"`. Possible values are

- `"numpy"` to use Numpy, or
- `"fftw"` to use pyFFTW, if installed.

The variable can be changed globally

```
udft.LIB = "numpy"
```

In addition, each function has a parameter to change the library used at call time. A `ValueError` is raided if the value is not recognized.

2.2 Discrete Fourier Transform

2.2.1 dftn

udft.**dftn**(*inarray*, *ndim=None*, *lib=None*)

ND unitary discrete Fourier transform.

Parameters

- **inarray** (*array-like*) – The array to transform.
- **ndim** (*int, optional*) – The *ndim* last axes along which to compute the transform. All axes by default.
- **lib** (*str, optional*) – Specify the library to compute the Fourier transform. See *set_lib get_lib* functions for the default.

Returns **outarray** – The DFT of *inarray* with same shape.

Return type array-like

2.2.2 idftn

udft.**idftn**(*inarray*, *ndim=None*, *lib=None*)

ND unitary inverse discrete Fourier transform.

Parameters

- **inarray** (*array-like*) – The array to transform.
- **ndim** (*int, optional*) – The *ndim* last axes along wich to compute the transform. All axes by default.
- **lib** (*str, optional*) – Specify the library to compute the Fourier transform. See *set_lib get_lib* functions for the default.

Returns **outarray** – The IDFT of *inarray* with same shape.

Return type array-like

2.2.3 dft

udft.**dft**(*inarray*, *lib=None*)

1D unitary discrete Fourier transform.

Compute the unitary DFT on the last axis.

Parameters

- **inarray** (*array-like*) – The array to transform.
- **lib** (*str, optional*) – Specify the library to compute the Fourier transform. See *set_lib get_lib* functions for the default.

Returns **outarray** – The DFT of *inarray* with same shape.

Return type array-like

2.2.4 idft

udft.**idft**(*inarray*, *lib=None*)

1D unitary inverse discrete Fourier transform.

Compute the unitary inverse DFT transform on the last axis.

Parameters

- **inarray** (*array-like*) – The array to transform.
- **lib** (*str, optional*) – Specify the library to compute the Fourier transform. See *set_lib get_lib* functions for the default.

Returns **outarray** – The DFT of *inarray* with same shape.

Return type array-like

2.2.5 dft2

udft.**dft2**(*inarray*, *lib=None*)

2D unitary discrete Fourier transform.

Compute the unitary DFT on the last 2 axes.

Parameters

- **inarray** (*array-like*) – The array to transform.
- **lib** (*str, optional*) – Specify the library to compute the Fourier transform. See *set_lib get_lib* functions for the default.

Returns **outarray** – The DFT of *inarray* with same shape.

Return type array-like

2.2.6 idft2

udft.**idft2**(*inarray*, *lib=None*)

2D unitary inverse discrete Fourier transform.

Compute the unitary IDFT on the last 2 axes.

Parameters

- **inarray** (*array-like*) – The array to transform.
- **lib** (*str, optional*) – Specify the library to compute the Fourier transform. See *set_lib get_lib* functions for the default.

Returns **outarray** – The IDFT of *inarray* with same shape.

Return type array-like

2.3 Real Discrete Fourier Transform

The transforms here suppose input of real values. In direct transform, the last transformed axis has length `N // 2 + 1`.

Note: The exact output shape for real transform can't be guessed from input shape. Therefor, only one inverse transform *irdftn()* is provided and that function ask for the output shape. The dimension *ndim* corresponds to the length of shape.

2.3.1 rdftn

udft.**rdftn**(*inarray*, *ndim=None*, *lib=None*)

ND real unitary discrete Fourier transform.

Consider the Hermitian property of output with input having real values.

Parameters

- **inarray** (*array-like*) – The array of real values to transform.

- **ndim** (*int, optional*) – The *ndim* last axes along which to compute the transform. All axes by default.
- **lib** (*str, optional*) – Specify the library to compute the Fourier transform. See *set_lib get_lib* functions for the default.

Returns **outarray** – The real DFT of *inarray* (the last axe as N // 2 + 1 length).

Return type array-like

2.3.2 irdftn

udft.**irdftn**(*inarray*, *shape*, *lib=None*)

ND real unitary inverse discrete Fourier transform.

Consider the Hermitian property of input and return real values.

Parameters

- **inarray** (*array-like*) – The array of complex values to transform.
- **shape** (*tuple of int*) – The output shape of the *len(shape)* last axes. The transform is applied on the *n=len(shape)* axes.
- **lib** (*str, optional*) – Specify the library to compute the Fourier transform. See *set_lib get_lib* functions for the default.

Returns **outarray** – The real IDFT of *inarray*.

Return type array-like

2.3.3 rdft

udft.**rdft**(*inarray*, *lib=None*)

1D real unitary discrete Fourier transform.

Compute the unitary real DFT on the last axis. Consider the Hermitian property of output with input having real values.

Parameters

- **inarray** (*array-like*) – The array to transform.
- **lib** (*str, optional*) – Specify the library to compute the Fourier transform. See *set_lib get_lib* functions for the default.

Returns **outarray** – The real DFT of *inarray*, where the last dim has length N//2+1.

Return type array-like

2.3.4 rdft2

udft.**rdft2**(*inarray*, *lib=None*)

2D real unitary discrete Fourier transform.

Compute the unitary real DFT on the last 2 axes. Consider the Hermitian property of output when input has real values.

Parameters

- **inarray** (*array-like*) – The array to transform.
- **lib** (*str, optional*) – Specify the library to compute the Fourier transform. See *set_lib get_lib* functions for the default.

Returns **outarray** – The real DFT of *inarray*, where the last dim has length N//2+1.

Return type array-like

2.4 Convolution related

2.4.1 ir2fr

udft.**ir2fr**(*imp_resp*, *shape*, *origin=None*, *real=True*)

Compute the frequency response from impulse responses.

This function makes the necessary correct zero-padding, zero convention, correct DFT etc.

The DFT is performed on the last *len(shape)* dimensions for efficiency (C-order array). Use numpy implementation.

Parameters

- **imp_resp** (*array-like*) – The impulse responses.
- **shape** (*tuple of int*) – A tuple of integer corresponding to the target shape of the frequency responses, without hermitian property. The DFT is performed on the *len(shape)* last axes of ndarray.
- **origin** (*tuple of int, optional*) – The index of the origin (0 coordinate) of the impulse response. The center of the array by default (*shape[i] // 2*).
- **real** (*boolean, optional*) – If True, *imp_resp* is supposed real, and real DFT is used.

Returns **out** – The frequency responses of shape *shape* on the last *len(shape)* dimensions. If *real* is *True*, the last dimension as lenght *N//2+1*.

Return type array-like

Notes

- The output is returned as C-contiguous array.
- For convolution, the result must be used with unitary discrete Fourier transform for the signal (*udftn* or equivalent).

2.4.2 fr2ir

udft.**fr2ir**(*freq_resp*, *shape*, *origin=None*, *real=True*)

Return the impulse responses from frequency responses.

This function makes the necessary correct zero-padding, zero convention, correct DFT etc. to compute the impulse responses from frequency responses.

The IR array is supposed to have the origin in the middle of the array.

The Fourier transform is performed on the last *len(shape)* dimensions for efficiency (C-order array). Use *np.fft*.

Parameters

- **freq_resp** (*array-like*) – The frequency responses.
- **shape** (*tuple of int*) – Output shape of the impulse responses.
- **origin** (*tuple of int, optional*) – The index of the origin (0, 0) of output the impulse response. The center by default (shape[i] // 2).
- **real** (*boolean, optional*) – If True, imp_resp is supposed real, and real DFT is used.

Returns **out** – The impulse responses of shape *shape* on the last *len(shape)* axes.

Return type array-like

Notes

- The output is returned as C-contiguous array.
- For convolution, the result has to be used with unitary discrete Fourier transform for the signal (udftn or equivalent).

2.4.3 diff_ir

udft.**diff_ir**(*ndim=1*, *axis=0*, *norm=False*)

Return the impulse response of first order differences.

Parameters

- **ndim** (*int, optional*) – The number of dimensions of the array on which the diff will apply.
- **axis** (*int, optional*) – The axis (dimension) where the diff operates.

Returns **out** – The impulse response

Return type array_like

2.4.4 laplacian

udft.**laplacian**(*ndim*, *norm=False*)

Return the Laplacian impulse response.

The second-order difference in each axes.

Parameters **ndim** (*int*) – The dimension of the Laplacian.

Returns **out** – The impulse response

Return type array_like

2.5 Other

2.5.1 hnorm

udft.**hnorm**(*inarray*, *inshape*)

Hermitian l2-norm of array in discrete Fourier space.

Compute the l2-norm of complex array

$$\|x\|_2 = \sqrt{\sum_{n=1}^{N} |x_n|^2}$$

considering the Hermitian property. Must be used with *rdftn*. Equivalent of *np.linalg.norm* for array applied on full Fourier space array (those obtained with *dftn*).

Parameters

- **inarray** (*array-like of shape (.. + inshape)*) – The input array with half of the Fourier plan.
- **inshape** (*tuple of int*) – The shape of the original array *oarr* where *inarray=rdft(oarr)*.

Returns **norm**

Return type float

2.5.2 crandn

Warning: At that time, this function does not respect the hermitian property, this is planned.

udft.**crandn**(*shape*)

Draw from white complex Normal.

Draw unitary DFT of real white Gaussian field of zero mean and unit variance. Does not consider hermitian property, *shape* is supposed to consider half of the frequency plane already.

Return type ndarray

INDEX