Sharp Documentation

Release 1.0

Tomas Fiers

Apr 25, 2019

Code documentation

CHAPTER 1

sharp

1.1 sharp package

1.1.1 Subpackages

sharp.config package

Subpackages

sharp.config.default package

Submodules

sharp.config.default.channels module

class sharp.config.default.channels.**Channel**(*name: str*, *index: int*, *recording_site: Union[str, NoneType] = None*, *x: Union[float, NoneType] = None*, *y: Union[float, NoneType] = None*)

Bases: object

recording_site = None

x = None

y = None

sharp.config.default.tasks module

Submodules

sharp.config.load module

sharp.config.spec module

sharp.data package

Classes that describe the data processed and produced by this package.

The actual data is stored somewhere else on the file system (as specified in *luigi.toml*).

Calculations performed in this module should be minimal (e.g property accesses should return near-instantaneously).

Subpackages

sharp.data.files package

Classes that describe the files processed by this software, and that provide an easy interface over them.

Submodules

sharp.data.files.base module

sharp.data.files.evaluation module

sharp.data.files.figure module

sharp.data.files.neuralnet module

sharp.data.files.neuralynx module

sharp.data.files.numpy module

sharp.data.files.stdlib module

sharp.data.hardcoded package

Subpackages

sharp.data.hardcoded.filters package

Submodules

sharp.data.hardcoded.filters.base module

sharp.data.hardcoded.filters.best module

sharp.data.hardcoded.filters.literature module

sharp.data.hardcoded.filters.util module

Submodules

sharp.data.hardcoded.style module

sharp.data.types package

Custom Python classes to encapsulate calculation results.

Mostly wrappers and extensions of standard Python datatypes and of datatypes from the dependencies of this package.

Subpackages

sharp.data.types.evaluation package

Submodules

sharp.data.types.evaluation.sweep module

sharp.data.types.evaluation.threshold module

Submodules

sharp.data.types.aliases module

sharp.data.types.intersection module

sharp.data.types.neuralnet module

sharp.data.types.signal module

sharp.data.types.slice module

sharp.data.types.split module

sharp.tasks package

A collection of *Luigi* tasks to describe entire raw data-to-figure pipelines.

These are batch jobs that take files as inputs and write files as outputs. Luigi resolves the dependencies between these tasks, and runs them, skipping tasks that have already been completed.

See also https://luigi.readthedocs.io

Subpackages

sharp.tasks.evaluate package

Submodules

sharp.tasks.evaluate.multi_envelope module

sharp.tasks.evaluate.sweep module

sharp.tasks.evaluate.threshold module

sharp.tasks.multilin package

LSM = Linear signal-to-noise ratio maximiser

Submodules

sharp.tasks.multilin.apply module

sharp.tasks.multilin.base module

sharp.tasks.multilin.train module

sharp.tasks.neuralnet package

Submodules

sharp.tasks.neuralnet.apply module

sharp.tasks.neuralnet.base module

sharp.tasks.neuralnet.select module

sharp.tasks.neuralnet.train module

sharp.tasks.neuralnet.util module

sharp.tasks.plot package

Note: the plots created in these modules do not plot data outside of the x-limits. This confines interactive exploration (e.g. in Jupyter Notebooks with *%matplotlib notebook*) to the immediately plotted data. It greatly speeds up figure creation however, which is useful for the long-duration signals that we deal with in this thesis.

Python Module Index

S

Index