skfda.representation.FData

class **skfda.representation.FData**(*extrapolation*, *dataset_label*, *axes_labels*, *keepdims*)
[source]

Defines the structure of a functional data object.

nsamples

Number of samples.

Type: int

ndim_domain

Dimension of the domain.

Type: int

ndim_image

Dimension of the image.

Type: int

extrapolation

Default extrapolation mode.

Type: Extrapolation

dataset_label

name of the dataset.

Type: str

axes_labels

list containing the labels of the different axis. The first element is the x label, the second the y label and so on.

Type: list

`keepdims`

Default value of argument keepdims in `evaluate()`.

Type: bool

`__init__`(*extrapolation*, *dataset_label*, *axes_labels*, *keepdims*) [source]

Initialize self. See help(type(self)) for accurate signature.

Methods

`__init__`(extrapolation, dataset_label, ...)	Initialize self.
`argsort`([ascending, kind])	Return the indices that would sort this array.
`astype`(dtype[, copy])	Cast to a NumPy array with 'dtype'.
`compose`(fd, *[, eval_points])	Composition of functions.
`concatenate`(other)	Join samples from a similar FData object.
`copy`(**kwargs)	Make a copy of the object.
`derivative`([order])	Differentiate a FData object.
`dropna`()	Return ExtensionArray without NA values
`evaluate`(eval_points, *[, derivative, ...])	Evaluate the object or its derivatives at a list
`factorize`([na_sentinel])	Encode the extension array as an enumerate
`fillna`([value, method, limit])	Fill NA/NaN values using the specified meth
`generic_plotting_checks`([fig, ax, nrows, ncols])	Check the arguments passed to both `plot` a
`isna`()	A 1-D array indicating if each value is missin
`mean`()	Compute the mean of all the samples.
`plot`([chart, derivative, fig, ax, nrows, ...])	Plot the FDatGrid object.
`repeat`(repeats[, axis])	Repeat elements of a ExtensionArray.
`searchsorted`(value[, side, sorter])	Find indices where elements should be inser
`set_figure_and_axes`(nrows, ncols)	Set figure and its axes.
`set_labels`([fig, ax, patches])	Set labels if any.

`shift` (shifts, *[, restrict_domain, ...])	Perform a shift of the curves.
`take` (indices[, allow_fill, fill_value])	Take elements from an array.
`to_basis` (basis[, eval_points])	Return the basis representation of the object
`to_grid` ([eval_points])	Return the discrete representation of the obj
`to_numpy` ()	Returns a numpy array with the objects
`unique` ()	Compute the ExtensionArray of unique value

Attributes

`domain_range`	Return the domain range of the object
`dtype`	An instance of 'ExtensionDtype'.
`extrapolation`	Return default type of extrapolation.
`extrapolator_evaluator`	Return the evaluator constructed by the extrapolator.
`nbytes`	The number of bytes needed to store this object in memory.
`ndim`	Return number of dimensions of the functional data.
`ndim_codomain`	Return number of dimensions of the codomain.
`ndim_domain`	Return number of dimensions of the domain.
`ndim_image`	Return number of dimensions of the image.
`nsamples`	Return the number of samples.
`shape`	Return a tuple of the array dimensions.