skfda.misc.metrics.vectorial_norm

skfda.misc.metrics.vectorial_norm(*fdatagrid*, *p=2*) [source]

Apply a vectorial norm to a multivariate function.

Given a multivariate function $f : \mathbb{R}^n \to \mathbb{R}^d$ applies a vectorial norm $\| \cdot \|$ to produce a function $\|f\| : \mathbb{R}^n \to \mathbb{R}$.

For example, let $f : \mathbb{R} \to \mathbb{R}^2$ be $f(t) = (f_1(t), f_2(t))$ and $\| \cdot \|_2$ the euclidian norm.

$$\|f\|_2(t) = \sqrt{|f_1(t)|^2 + |f_2(t)|^2}$$

In general if $p \neq \pm\infty$ and $f : \mathbb{R}^n \to \mathbb{R}^d$

$$\|f\|_p(x_1, \ldots x_n) = \left(\sum_{k=1}^{d} |f_k(x_1, \ldots, x_n)|^p \right)^{(1/p)}$$

Parameters:

- **fdatagrid** (`FDatagrid`) – Functional object to be transformed.
- **p** (*int*, *optional*) – Exponent in the lp norm. If p is a number then it is applied sum(abs(x)**p)**(1./p), if p is inf then max(abs(x)), and if p is -inf it is applied min(abs(x)). See numpy.linalg.norm to more information. Defaults to 2.

Returns: FDatagrid with image dimension equal to 1.

Return type: (`FDatagrid`)

Examples

```
>>> from skfda.datasets import make_multimodal_samples
>>> from skfda.misc.metrics import vectorial_norm
```

First we will construct an example dataset with curves in $\mathbb{R}^2$.

```
>>> fd = make_multimodal_samples(ndim_image=2, random_state=0)
>>> fd.ndim_image
2
```

We will apply the euclidean norm

```
>>> fd = vectorial_norm(fd, p=2)
>>> fd.ndim_image
1
```