

Shift Registration of basis

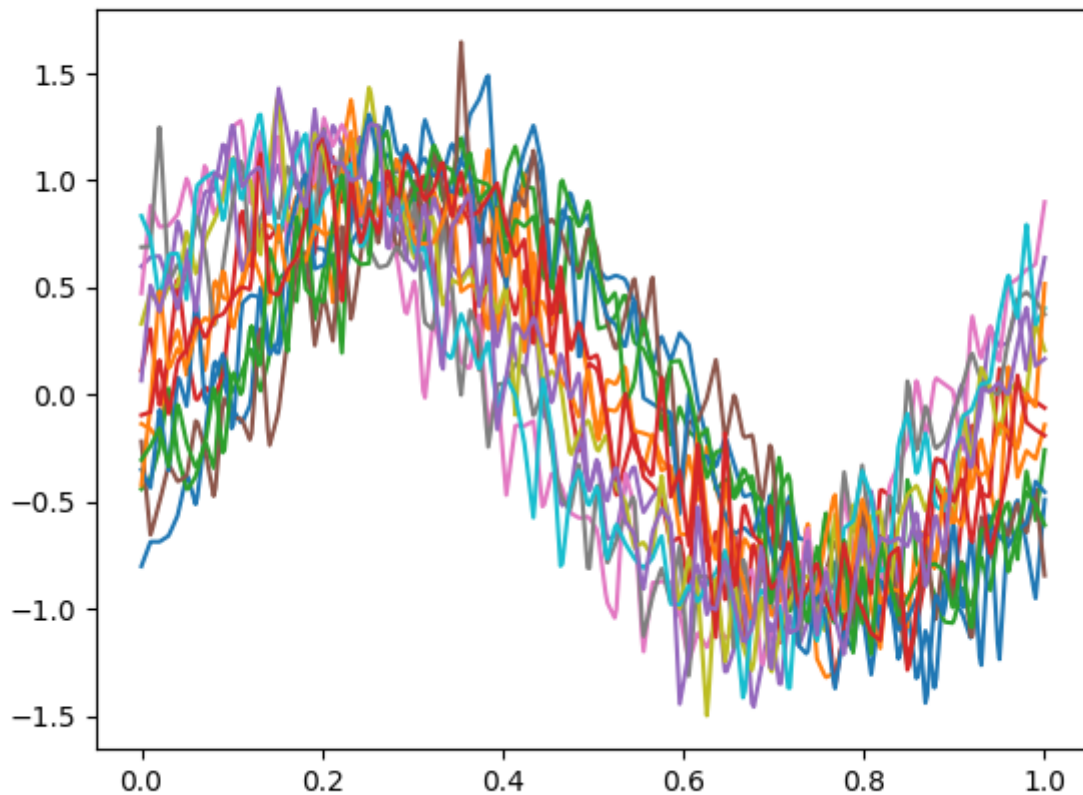
Shows the use of shift registration applied to a sinusoidal process represented in a Fourier basis.

```
# Author: Pablo Marcos Manchón  
# License: MIT  
  
# sphinx_gallery_thumbnail_number = 3  
  
import skfda  
import matplotlib.pyplot as plt
```

In this example we will use a `sinusoidal process` synthetically generated. This dataset consists in a sinusoidal wave with fixed period which contains phase and amplitude variation with gaussian noise.

In this example we want to register the curves using a translation and remove the phase variation to perform further analysis.

```
fd = skfda.datasets.make_sinusoidal_process(random_state=1)  
fd.plot()
```

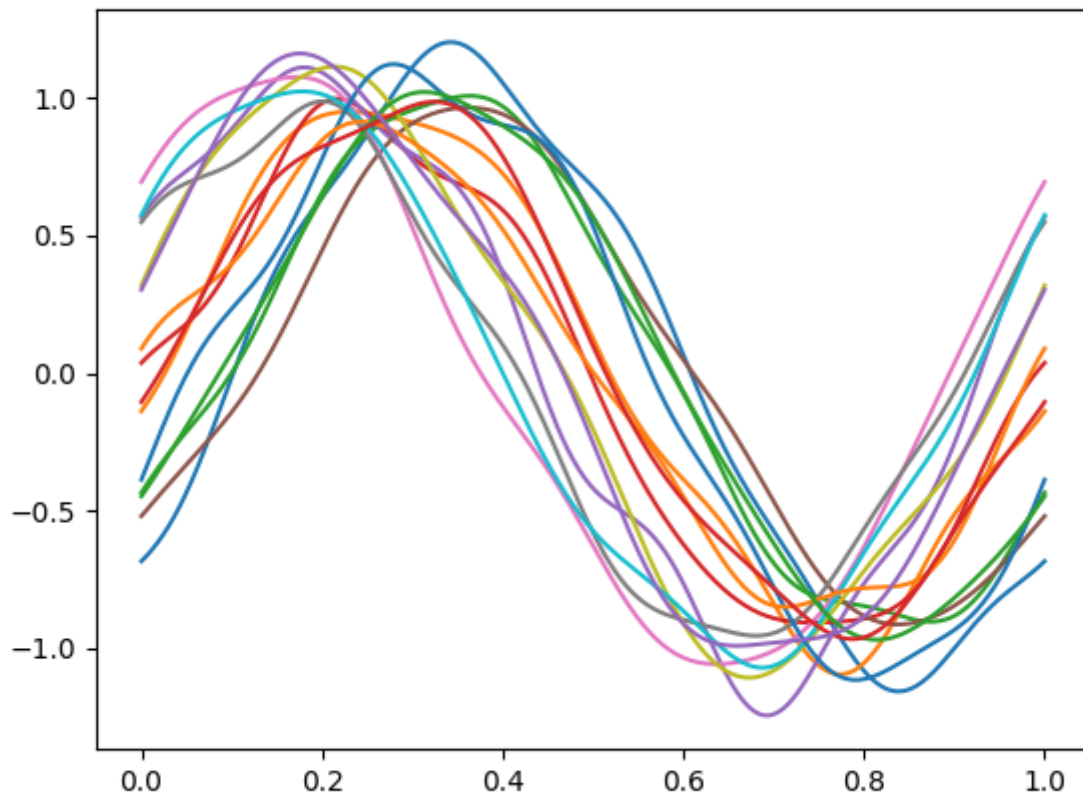


We will smooth the curves using a basis representation, which will help us to remove the gaussian noise. Smoothing before registration is essential due to the use of derivatives in the optimization process.

Because of their sinusoidal nature we will use a Fourier basis.

```
basis = skfda.representation.basis.Fourier(nbasis=11)
fd_basis = fd.to_basis(basis)

plt.figure()
fd_basis.plot()
```

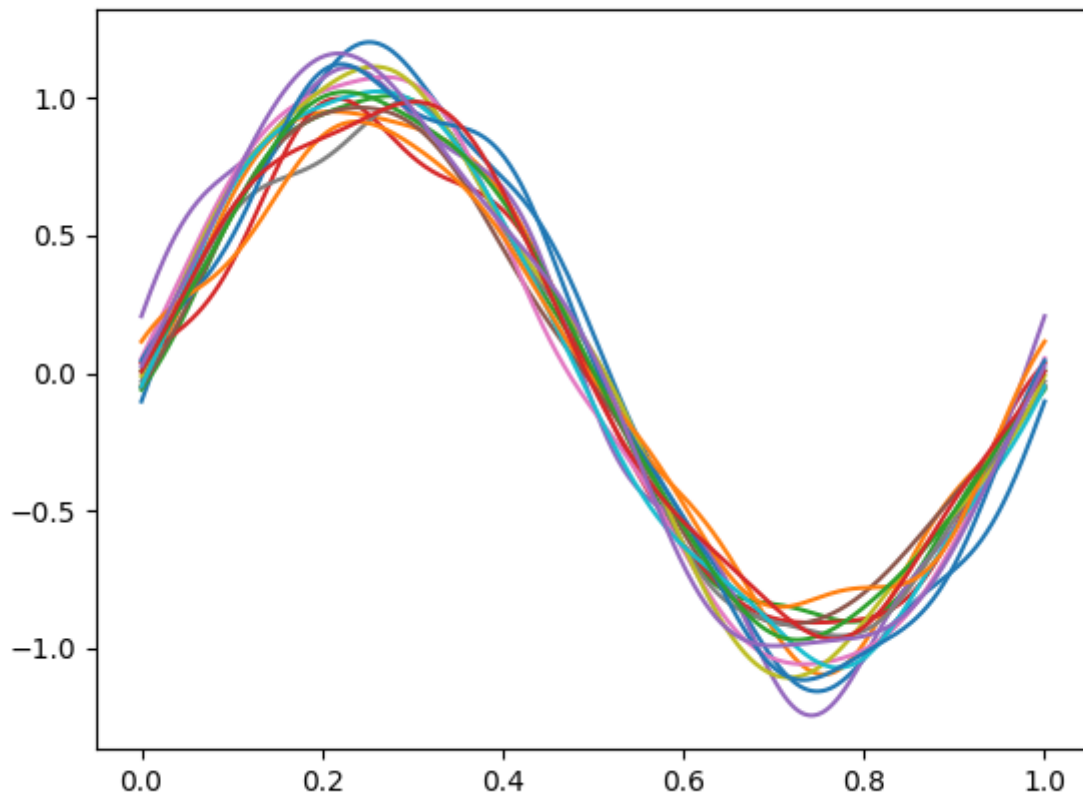


We will apply the `shift_registration`, which is suitable due to the periodicity of the dataset and the small amount of amplitude variation.

```
fd_registered = skfda.preprocessing.registration.shift_registration(fd_basis)
```

We can observe how the sinusoidal pattern is easily distinguishable once the alignment has been made.

```
plt.figure()
fd_registered.plot()
```



We will plot the mean of the original smoothed curves and the registered ones, and we will compare with the original sinusoidal process without noise.

We can see how the phase variation affects to the mean of the original curves varying their amplitude with respect to the original process, however, this effect is mitigated after the registration.

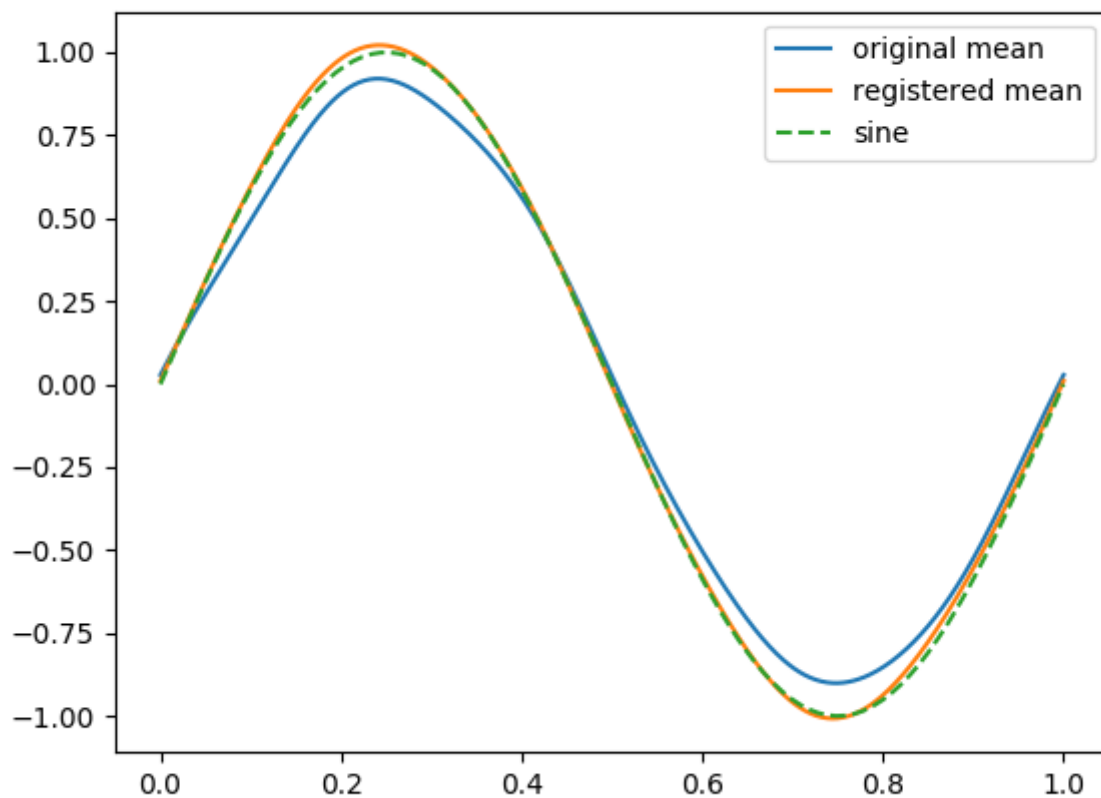
```
plt.figure()

fd_basis.mean().plot()
fd_registered.mean().plot()

# sinusoidal process without variation and noise
sine = skfda.datasets.make_sinusoidal_process(n_samples=1, phase_std=0,
                                              amplitude_std=0, error_std=0)

sine.plot(linestyle='dashed')

plt.legend(['original mean', 'registered mean', 'sine'])
```



The values of the shifts δ_i may be relevant for further analysis, as they may be considered as nuisance or random effects.

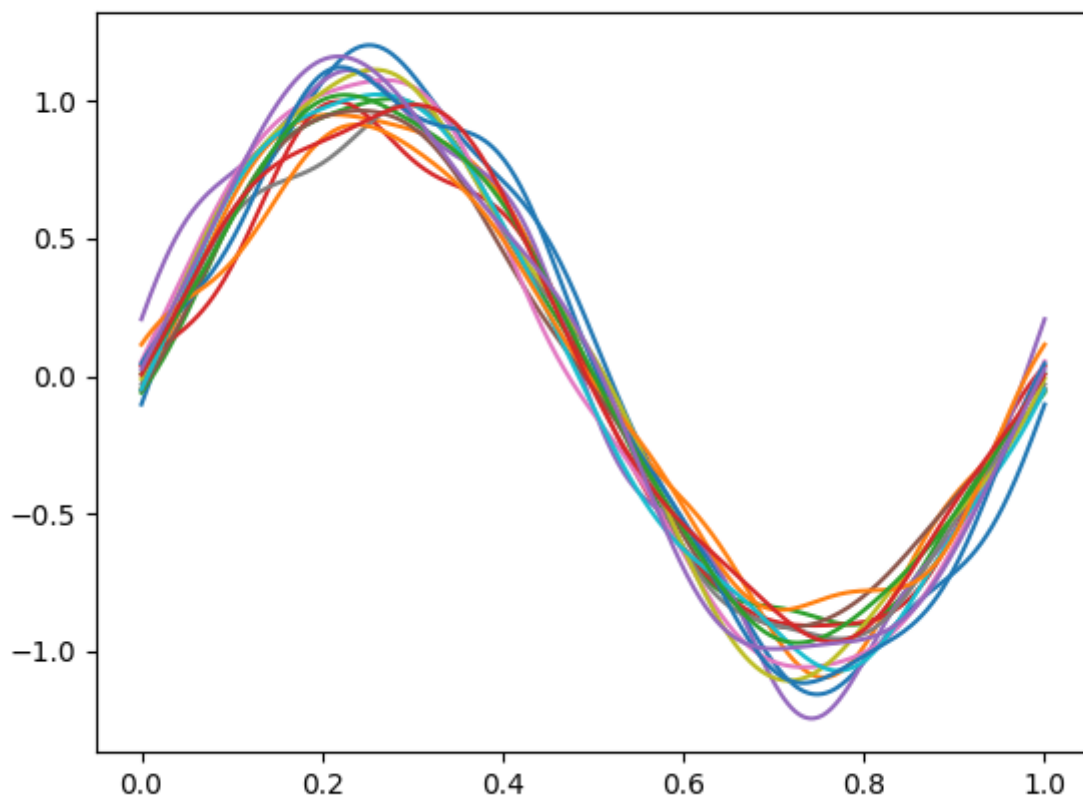
```
deltas = skfda.preprocessing.registration.shift_registration_deltas(fd_basis)
print(deltas)
```

Out:

```
[ 0.09004943  0.01808744  0.08732826 -0.00013559 -0.04950421  0.11984576
 -0.09723283 -0.09330286 -0.04398832 -0.08389279  0.0583045   0.00503724
 0.08788296  0.0214795  -0.042531   ]
```

The aligned functions can be obtained from the δ_i list using the *shift* method.

```
fd_basis.shift(deltas).plot()
```



Total running time of the script: (0 minutes 0.817 seconds)

Gallery generated by Sphinx-Gallery