Investigating Mean Integrated Squared Error calculation in R

Tahmidul Islam

7/13/2020

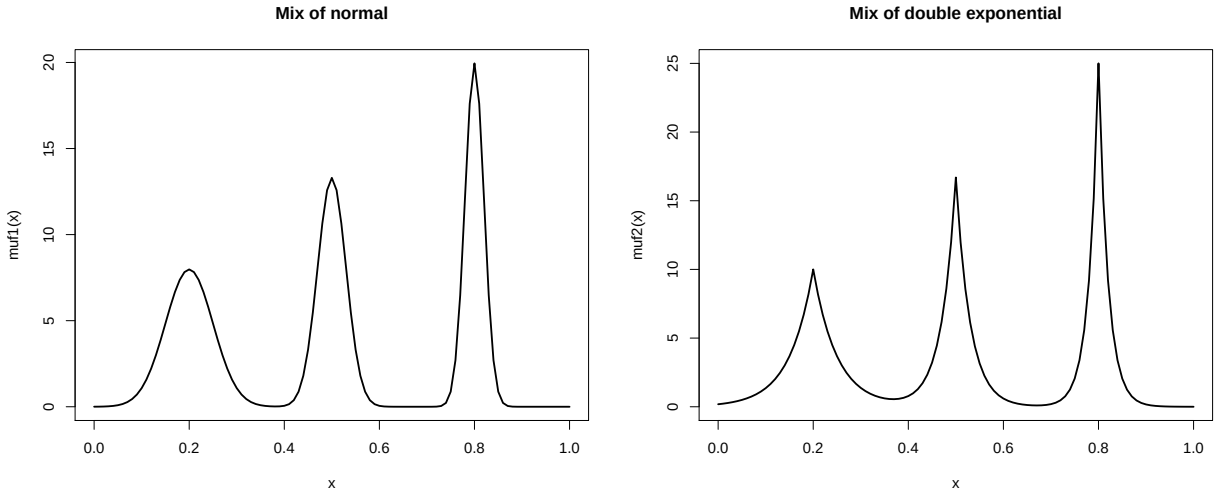
We propose two new mean functions to be used for functional data simulation. One is based on normal density functions with different location and scale parameters. The other is based on Laplace (double exponential) distribution with similar manipulation. These two mean functions contain spikes of several degrees (depending on the scale parameter). The Laplace distribution outputs non-differentiable regions at the spikes. Proposed new mean function:

$$\mu_1(t) = \ N(.2, .05) + \ N(.3, .03) + \ N(.8, .02); \ t \in (0, 1).$$

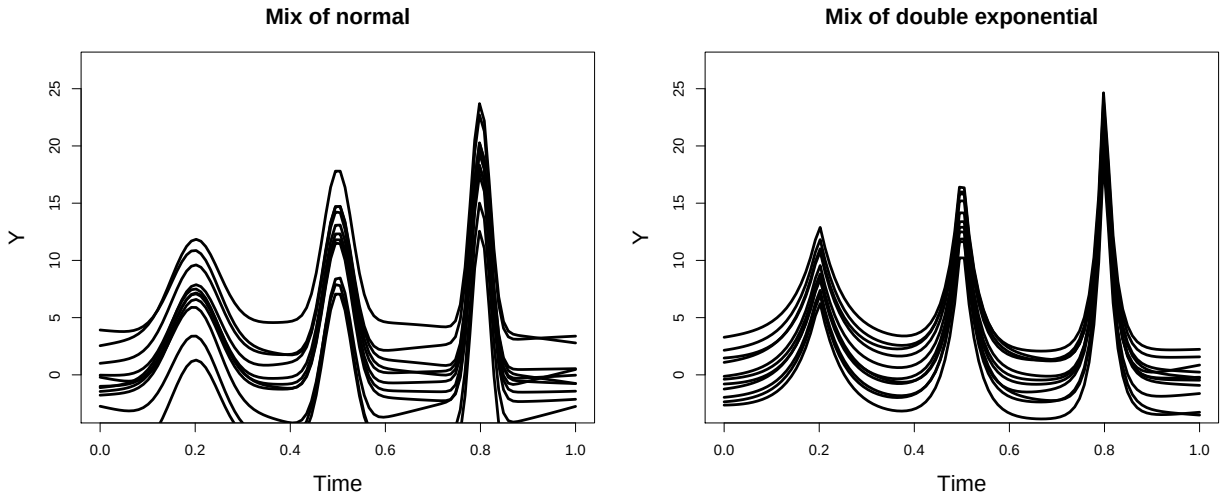
$$\mu_2(t) = \ dexp(.2, .05) + \ dexp(.3, .03) + \ dexp(.8, .02); \ t \in (0, 1).$$

Where $N(a, b)$ and $dexp(a, b)$ are the density function of a normal distribution and a Laplace distribution respectively with location parameter a and scale parameter b.

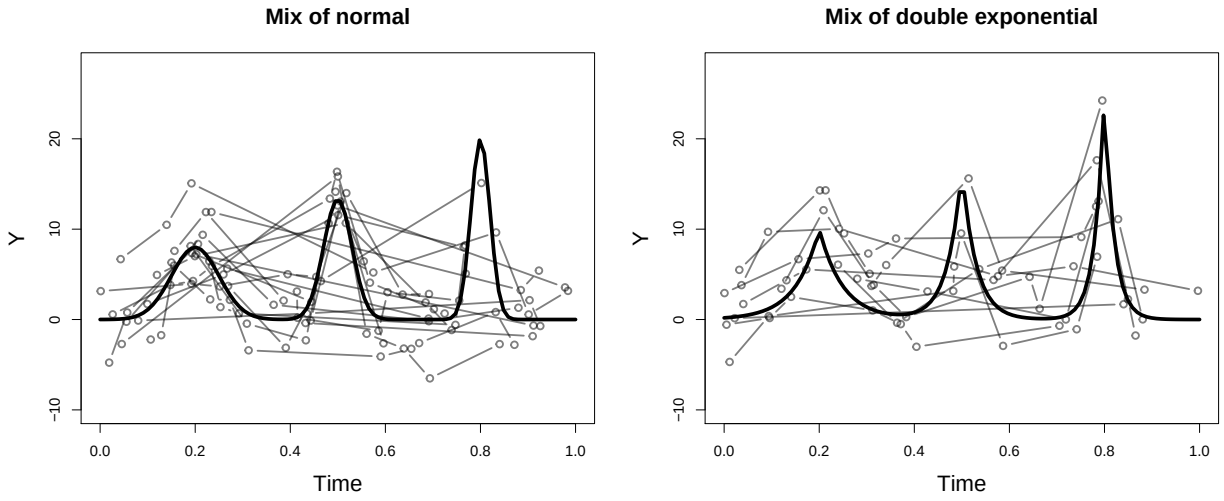
Let us plot the mean funtions.



Now simulate some random functions using the mean functions as the prior mean in the Gaussian process with Gaussian covariance function.



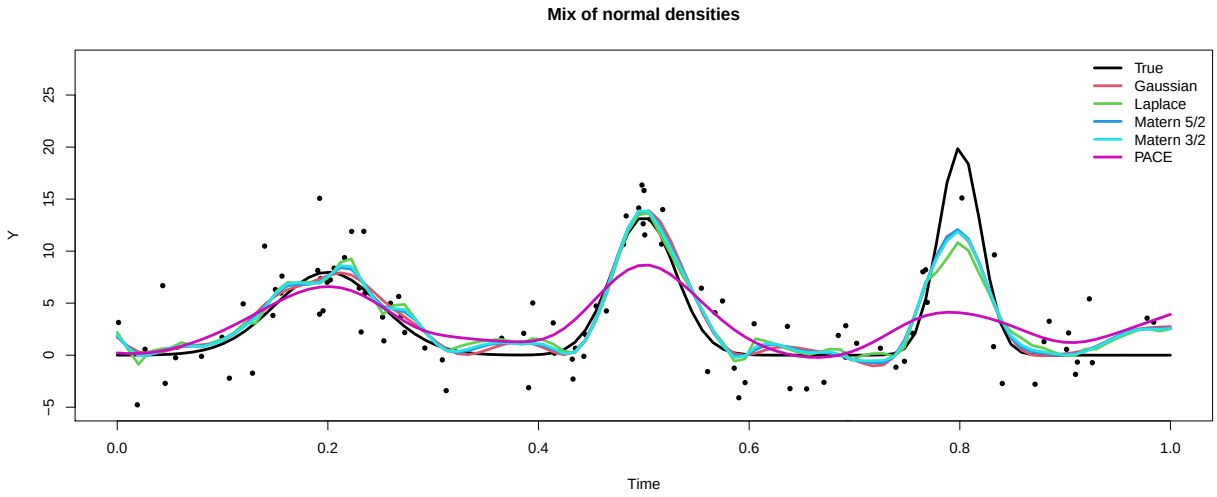
Now we generate sparse functional data which are actually observed in practice. To introduce sparsity, first we sample any integer n_t between 2 to 10. This is the number of time points each function will be observed on. Further we sample n_t time points from a uniform (0, 1).



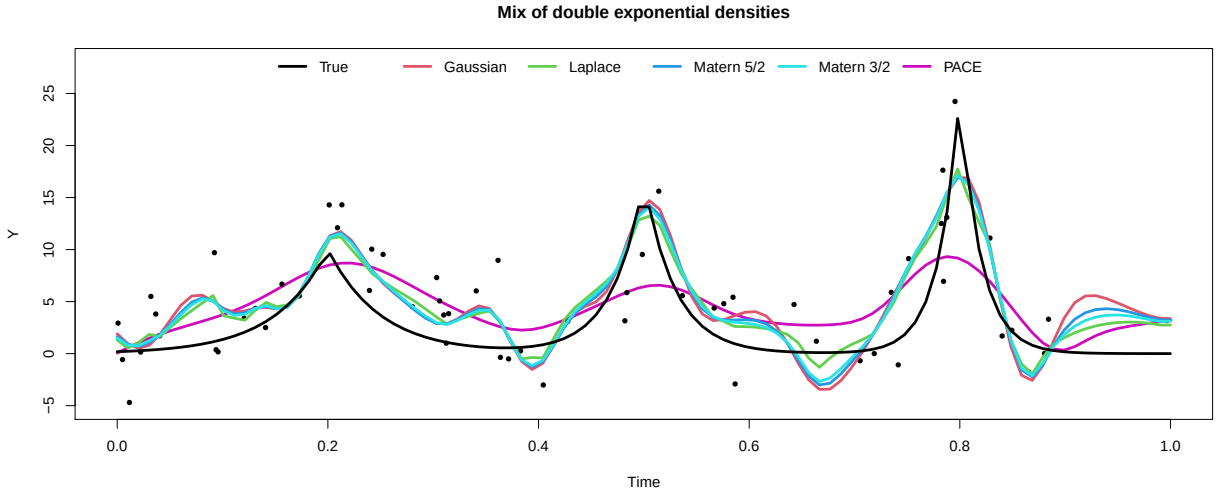
We will fit our GP based model to the generated data and compare with the fit with the PACE (Yao, Mueller, and Wang (2005)) method based on Functional Principal Component Analysis (FPCA), a key technique for functional data analysis, for sparsely or densely sampled random trajectories and time courses, via the Principal Analysis by Conditional Estimation (PACE) algorithm.

Mix of normal density mean function

We can inspect the fit of the mean function estimation by GP based method and compare with PACE. For GP method, we have used four different covariance kernels: Gaussian (RBF), Laplace, Matern 5/2 and Matern 3/2.



We can repeat this procedure for the mix of double exponential density mean function.



Error in mean function estimation

Standardized Average Squared Error (SASE)

We want to measure the goodness of fit of our GP based method and compare the performance with PACE. The basic error metric we can use for mean function estimation is the Standardized Average Squared Error (SASE). SASE is calculated by taking a grid of equally spaced time points, computing the mean of the squared estimation errors. It is then standardized by the variance of the responses. Since we know the true mean function, we can compute the following:

$$\text{ASE} = \frac{1}{N}\sum_{i=1}^{N}\Big(\hat{\mu}(t_i) - \mu(t_i)\Big)^2.$$

$$SASE = ASE/var(y)$$

Here the time points t_i's form a fixed grid of points over the support of the function. Here we can choose 10000 time points from the interval $(0, 1)$ and compute the SASE.

Method	Gaussian	Laplace	Matern_5/2	Matern 3/2	PACE
Mix of normal	0.0935	0.1265	0.0916	0.0962	0.3789
Mix of double exponential	0.3129	0.2076	0.2662	0.2431	0.2978

Mean Integrated Squared Error (MISE)

The mean integrated squared error (MISE) is defined by

$$E||\hat{f} - f||^2 = E \int \left(\hat{f}(t) - f(t)\right)^2 dt.$$

Where f is the true function. We first compute the integrated squared error (ISE). To do that, we use numerical integration of the squared error function over $(0, 1)$. The ISE is then standardized using the variance of the response.

Method	Gaussian	Laplace	Matern_5/2	Matern 3/2	PACE
Mix of normal	0.0934	0.1265	0.0916	0.0962	0.3789
Mix of double exponential	0.344	0.2282	0.2926	0.2673	0.3274

Since R's base code for integration using adaptive quadrature sometimes results in error, we use a package called 'cubature'. This package uses adative and monte-carlo integration. We show results for the mix of normal density mean function.

Method	Gaussian	Laplace	Matern_5/2	Matern 3/2	PACE
ISE: Mix of normal	0.0934	0.1265	0.0916	0.0962	0.3789
ISE: Mix of double exponential	0.344	0.2282	0.2927	0.2673	0.3274

We can also approximate the integration by Monte Carlo method. The simplest version would be to use the uniform distribution. We generate $N = 10000$ time points from uniform(0,1) and approximate the integral.

$$\int_a^b f(t)dt = (b-a)\frac{1}{N}\sum f(t_i).$$

Method	Gaussian	Laplace	Matern_5/2	Matern 3/2	PACE
ISE: Mix of normal	0.0937	0.1258	0.0934	0.0974	0.9205
ISE: Mix of double exponential	0.3425	0.2286	0.2917	0.2664	0.7927

Summary of error calculation

Mix normal mean function

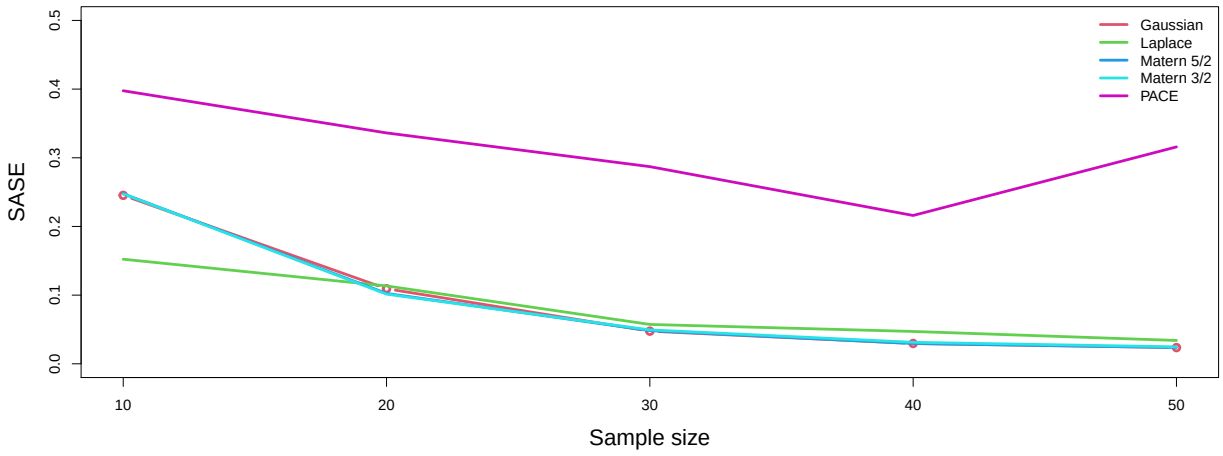
Method	Gaussian	Laplace	Matern_5/2	Matern 3/2	PACE
SASE	0.0935	0.1265	0.0916	0.0962	0.3789
ISE (R base function)	0.0934	0.1265	0.0916	0.0962	0.3789
ISE (Cubature package)	0.0934	0.1265	0.0916	0.0962	0.3789
ISE (MC)	0.0937	0.1258	0.0934	0.0974	0.9205

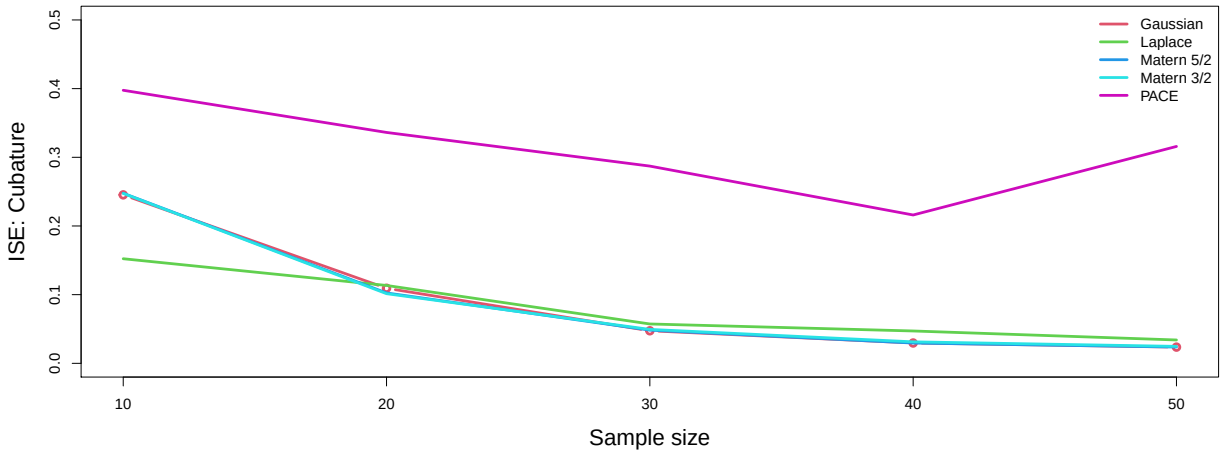
Mix double exponential mean function

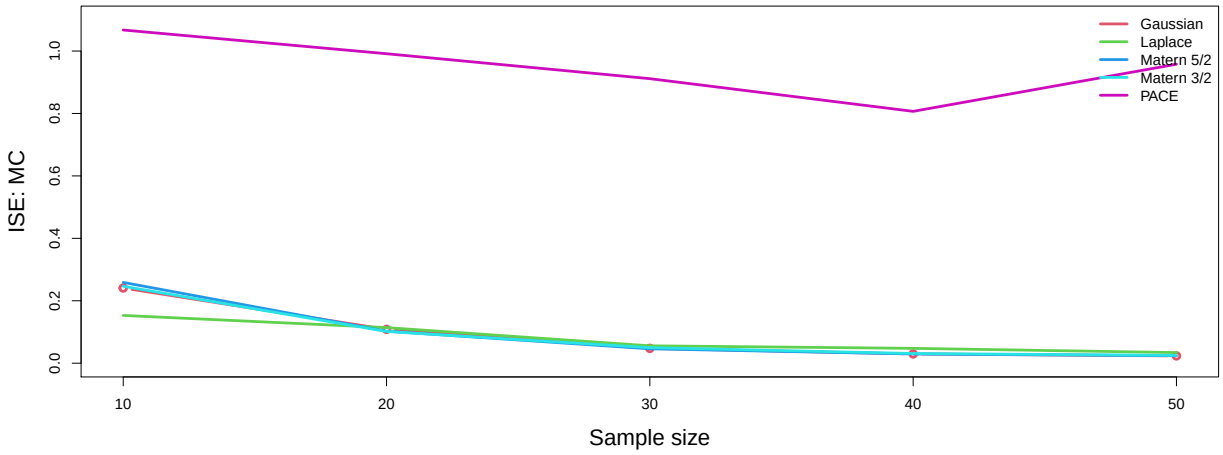
Method	Gaussian	Laplace	Matern_5/2	Matern 3/2	PACE
SASE	0.3129	0.2076	0.2662	0.2431	0.2978
ISE (R base function)	0.344	0.2282	0.2926	0.2673	0.3274
ISE (Cubature package)	0.344	0.2282	0.2927	0.2673	0.3274
ISE (MC)	0.3425	0.2286	0.2917	0.2664	0.7927

Effect of Sample Size

We wish to investigate how these error measurement changes if we change our sample size, i.e. the number of observed curves in the dataset. For this purpose, we vary the sample size $n = 10, 20, 30, 40$ and 50. For each of these sample sizes, we generate data using the first mean function and calculate the SASE, ISE based on Cubature package and ISE based on MC method. We plot the error metrics as a function of the sample sizes.






ISE: MC
0.0
0.2
0.4
0.6
0.8
1.0
10
20
30
40
50
Sample size
Gaussian
Laplace
Matern 5/2
Matern 3/2
PACE