Learning Mixtures of Gaussian Processes through Random Projection

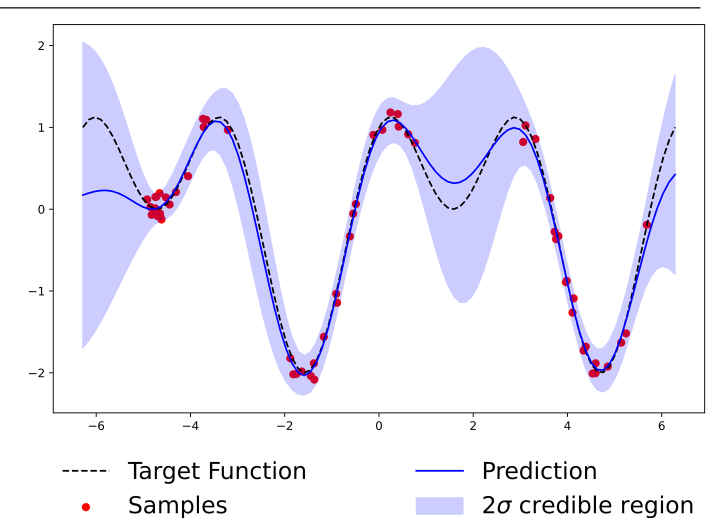
Emmanuel Akeweje[1], Mimi Zhang [1,2]

[1]School of Computer Science and Statistics, Trinity College Dublin [2]I-Form Advanced Manufacturing Research Centre, Science Foundation Ireland

Trinity College Dublin
Coláiste na Tríonóide, Baile Átha Cliath
The University of Dublin

INTRODUCTION

- A Gaussian process $GP(\mu, \Sigma)$ is a stochastic process: if $X \sim GP(\mu, \Sigma)$, then $\forall\ m$ and $\boldsymbol{t} = (t_1, \dots, t_m)^T$, $X(\boldsymbol{t}) \sim N(\mu(\boldsymbol{t}), \Sigma(\boldsymbol{t}, \boldsymbol{t}))$.
- A GP mixture is characterized by K Gaussian random functions $X_k \sim GP(\mu_k, \Sigma_k)$, $k = 1, \dots, K$, and the mixing proportions $\{\pi_k\}_{k=1}^K$.



Existing methods treat the mixture-learning problem as a parameter-estimation problem and develop EM-type algorithms. However, EM-type algorithms are of heavy computational load and highly sensitive to initialization.

Our approach performs cluster analysis before parameter estimation, and once the hidden cluster labels are revealed, the problem of learning the mixture of GPs reduces to learning each GP component independently.

PROJECTING ONTO ONE DIMENSION

Gaussian Random Variable

$$[X|Z = k] = X_k \sim N(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k):$$
$$\boldsymbol{\Sigma}_k = \sum_{r=1}^{\infty} \lambda_{kr} \boldsymbol{b}_{kr} \boldsymbol{b}_{kr}^T.$$

$\forall \boldsymbol{y} \in \mathbb{R}^p$,

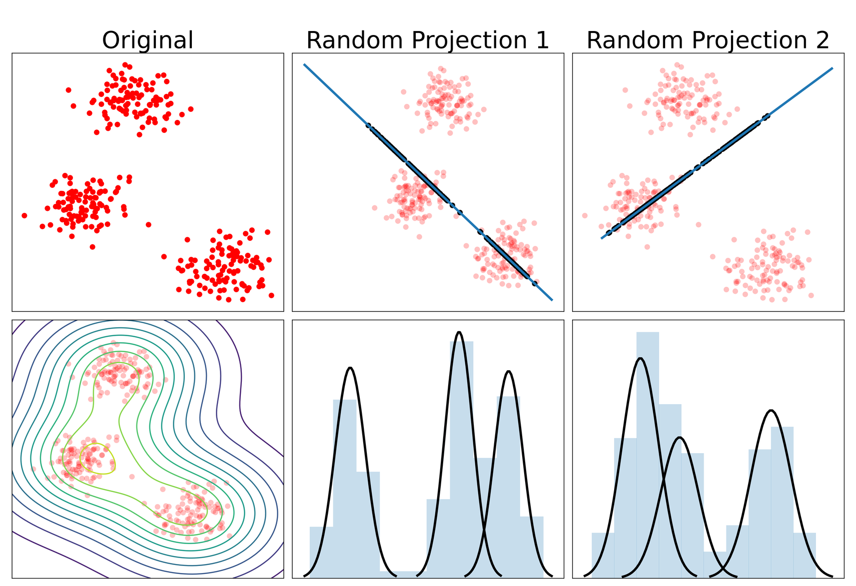
$$\langle X_k, \boldsymbol{y} \rangle \sim N(\langle \boldsymbol{\mu}_k, \boldsymbol{y} \rangle, \sum_{r=1}^{\infty} \lambda_{kr} \langle \boldsymbol{b}_{kr}, \boldsymbol{y} \rangle^2).$$
$$\langle X, \boldsymbol{y} \rangle \sim \sum_{k=1}^{K} \pi_k N(\langle \boldsymbol{\mu}_k, \boldsymbol{y} \rangle, \sum_{r=1}^{\infty} \lambda_{kr} \langle \boldsymbol{b}_{kr}, \boldsymbol{y} \rangle^2).$$

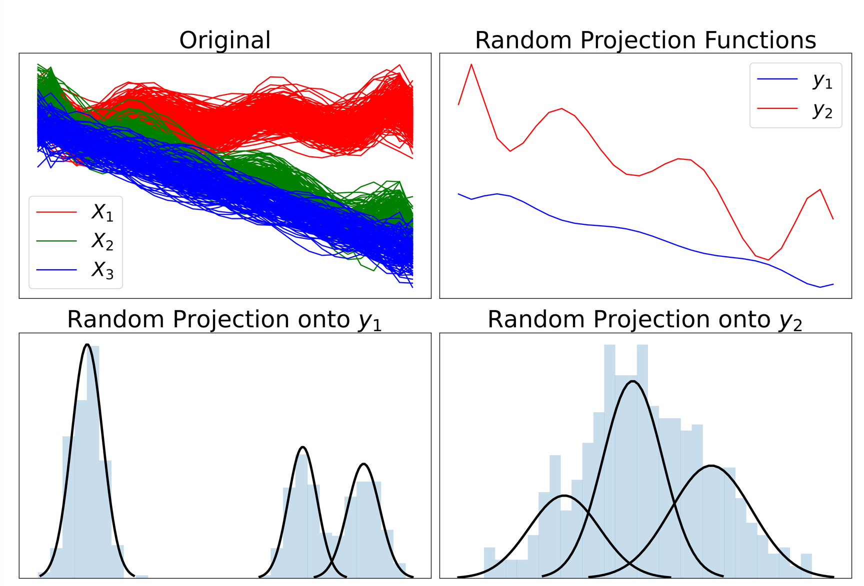
Gaussian Random Function

$$[X|Z = k] = X_k \sim GP(\mu_k, \Sigma_k):$$
$$\Sigma_k(s, t) = \sum_{r=1}^{\infty} \lambda_{kr} b_{kr}(s) b_{kr}(t).$$

$\forall y \in \mathcal{H}(\tau, \mathbb{R})$,

$$\langle X_k, y \rangle \sim N(\langle \mu_k, y \rangle, \sum_{r=1}^{\infty} \lambda_{kr} \langle b_{kr}, y \rangle^2).$$
$$\langle X, y \rangle \sim \sum_{k=1}^{K} \pi_k N(\langle \mu_k, y \rangle, \sum_{r=1}^{\infty} \lambda_{kr} \langle b_{kr}, y \rangle^2).$$





Wisdom of the Crowd

- Each projection function produces a GMM distribution of the projection coefficients, from which we can infer the hidden cluster labels.
- Different projection functions might reveal different structures in the data.
- Ensemble clustering encompasses all the information contained in the different GMMs and hence uncovers underlying patterns that might be missed by individual projections.
- Weight the individual projections according to the overlapping degree between the component Gaussian distributions.

METHODOLOGY

Algorithm 1 The GP mixture learning algorithm (GPmix)

Input: The raw data $\mathcal{D} = \{y_i(\underline{t}_i)\}_{i=1}^n$, the projection functions $\{\beta_v\}_{v=1}^V$, and the number of clusters K.

Output: The learned cluster labels $\{z_i\}_{i=1}^n$.

Estimate the population mean function μ and the n sample functions $\{x_i\}_{i=1}^n$.

from raw data to continuous functions

for $v = 1, \dots, V$ **do**

Calculate the n projection coefficients:

$$\alpha_{iv} = \langle x_i - \mu, \beta_v \rangle, \quad 1 \le i \le n.$$

Train a univariate GMM from the data $\{\alpha_{iv}\}_{i=1}^n$, denoted by

$$\sum_{k=1}^{K} \pi_{vk} \phi(\alpha; u_{vk}, \sigma_{vk}^2).$$

Obtain the cluster membership matrix $\mathbf{M}_v$:

$$m_{ik}^v = \frac{\pi_{vk} \phi(\alpha_{iv}; u_{vk}, \sigma_{vk}^2)}{\sum_{j=1}^K \pi_{vj} \phi(\alpha_{iv}; u_{vj}, \sigma_{vj}^2)}, \ 1 \le i \le n, \ \ 1 \le k \le K.$$

Construct a binary membership indicator matrix $\mathbf{B}_v$:

$$b_{ik}^v = \begin{cases} 1, \text{ if } k = \arg\max_{1 \le j \le K} \{m_{ij}^v\}; \\ 0, \text{ otherwise.} \end{cases}$$

Calculate the weight $w_v (> 0)$: $\sum_{v=1}^V w_v = 1$.

end for

projecting functional data & learning a univariate GMM from the projection coefficients

Apply a multivariate clustering method on the affinity matrix

$$\mathbf{A} = \sum_{v=1}^{V} w_v \mathbf{B}_v \mathbf{B}_v^T$$

and return the identified cluster labels $\{z_i\}_{i=1}^n$.

extracting a consensus clustering from the multiple GMMs

NUMERICAL STUDY

We assessed the effectiveness of the GPmix algorithm using 10 real datasets, comparing it to established functional data clustering methods such as fun**FEM**, fun**HDD**C, fun**clu**st, **FC**lust, **km**eans, and F**ADP**clust.

- Performance metrics include the Adjusted Rand Index (ARI) and Adjusted Mutual Information (AMI).
- The datasets, obtained from the UEA & UCR Time Series Classification Repository, are: ArrowHead (AH), BirdChicken (BC), CBF, DiatomSizeReduction (DSR), ECG200 (ECG), FaceFour (FF), GunPoint (GUP), Meat, Strawberry (SB), and Symbols (SYM).

Illustration of clustering procedure

We illustrate the clustering process using the CBF dataset. Figure 1 shows the projection functions, which are 14 wavelets from the Haar wavelet family. The sample curves are projected onto each of the Haar wavelets, producing 14 sets of projection coefficients. Figure 2 displays the distribution of the coefficients for each projection function. After learning individual GMMs, we calculate the weight for each GMM (namely, base clustering) according to the overlapping degree among the mixture components, as shown in Figure 3.

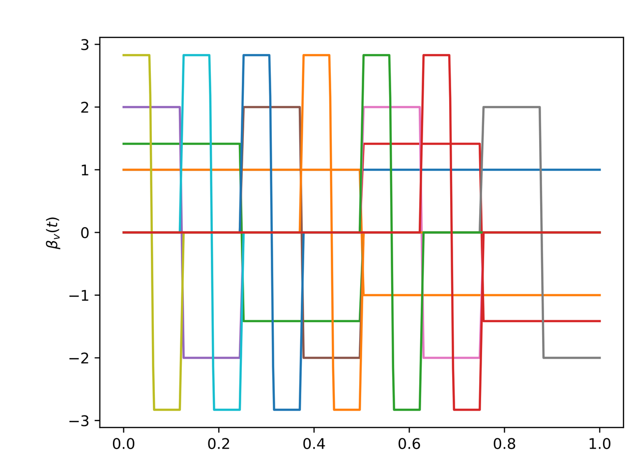


Figure 1. Haar wavelets as projection functions.

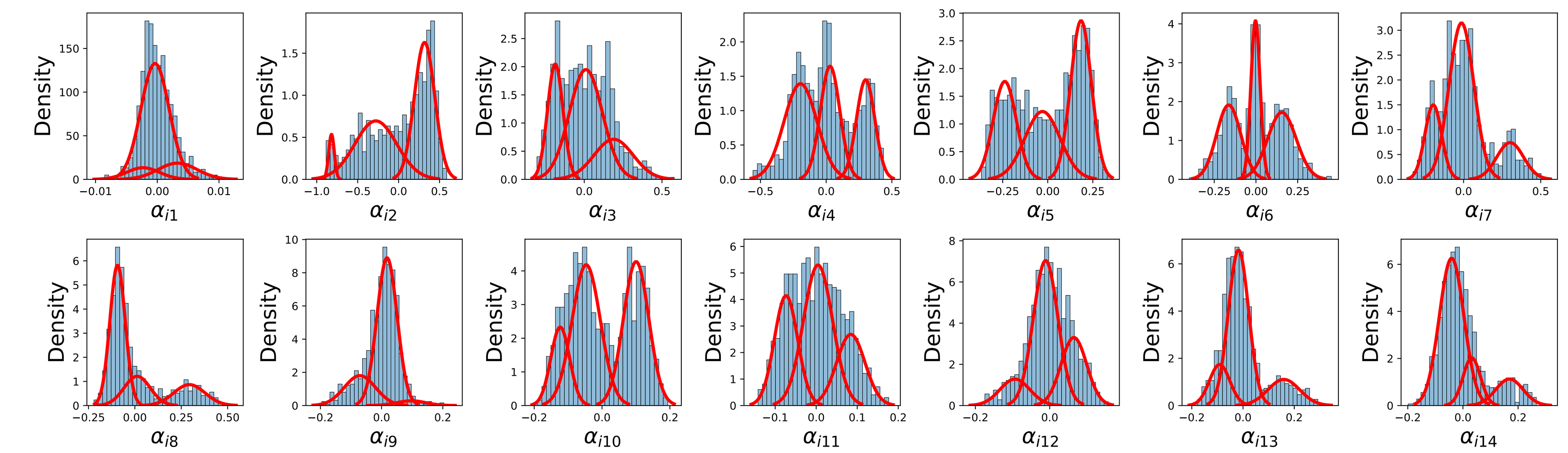


Figure 2. Histogram plots of the projection coefficients overlaid with the estimated density curves.

Table 1. AMI scores (upper line) and ARI scores (lower line) for the 10 real datasets.

Data	GPmix	FEM	HDD	clu	FC	km	ADP
AH	**0.37**	0.25	0.22	0.05	0.24	0.28	0.19
	0.36	0.29	0.21	0.01	0.25	0.26	0.18
BC	**0.24**	0.03	0.06	0.22	0.08	0.10	0.08
	0.29	0.04	0.07	0.15	0.10	0.10	0.10
CBF	**0.84**	0.37	0.47	0.01	0.53	0.34	0.40
	0.87	0.35	0.44	0.00	0.44	0.31	0.31
DSR	**0.94**	0.79	0.82	0.00	0.83	0.72	0.78
	0.95	0.83	0.86	0.01	0.86	0.73	0.82
ECG	**0.37**	0.15	0.17	0.03	0.37	0.17	0.07
	0.38	0.26	0.28	0.03	0.37	0.28	0.14
FF	**0.77**	0.47	0.40	0.06	0.56	0.50	0.44
	0.76	0.41	0.36	0.08	0.54	0.45	0.32
GUP	**0.34**	0.00	0.00	0.02	0.00	0.15	0.01
	0.25	0.00	0.00	0.02	0.00	0.07	0.01
Meat	0.70	**0.93**	0.54	0.36	0.54	0.66	0.72
	0.69	**0.95**	0.44	0.37	0.49	0.69	0.69
SB	**0.32**	0.08	0.00	0.03	0.12	0.07	0.03
	0.30	0.00	0.00	0.03	0.04	0.07	0.05
SYM	0.75	0.63	0.77	0.00	**0.85**	0.69	0.37
	0.66	0.53	0.67	0.00	**0.80**	0.62	0.30

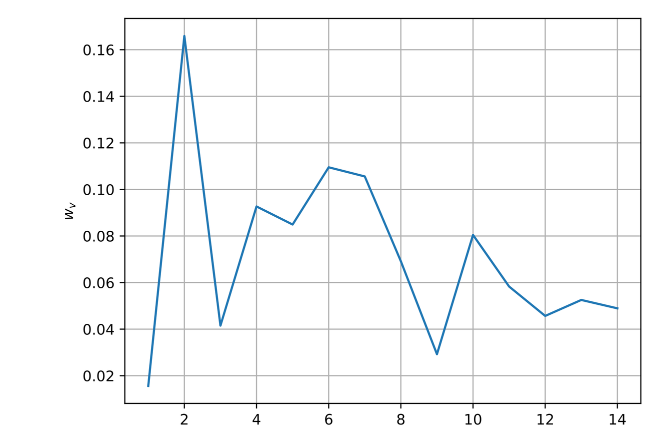


Figure 3. Base clustering weights.

Table 2. The runtime, in milliseconds, for the real datasets.

Data	GPmix	FEM	HDD	clu	FC	km	ADP
AH	107	120	**59**	2.1k	290k	215	396
BC	**18**	27	20	242	2m	66	45
CBF	**242**	2.2k	4.0k	102k	57k	893	5.9k
DSR	625	**481**	3.3k	6.3k	1.2m	507	1.0k
ECG	53	49	**37**	20k	14k	121	378
FF	**157**	413	147	390	1.4m	192	158
GUP	**50**	283	106	18k	47k	134	400
Meat	**58**	744	108	1548	2m	160	167
SB	**144**	1.6k	1.5k	1.6m	570k	683	14k
SYM	**551**	9.8k	4.0k	1.2m	3.4m	2.9k	40k

CONCLUSION

We developed a simple yet efficient technique for learning GP mixture models. Our method involves projecting functional data onto multiple one-dimensional functions, and learning a univariate GMM for each projection. We established a lower bound on the expected number of projections required to achieve effective separation within the 1-dimensional mixture components. Notably, our numerical study demonstrated the robust performance of our method even in cases where the functional data are not Gaussian.

Scan to access full paper and code:

ACKNOWLEDGEMENT

This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No 101002240) and funding from the Science Foundation Ireland under Grant number 21/RC/10295_P2.

Funded by the European Union
European Research Council
Established by the European Commission
Science Foundation Ireland For what's next