Multi-Kernel Probability Distribution Regressions

Pingping Zhu†, Hongchuan Wei‡, Wenjie Lu‡, and Silvia Ferrari†
Laboratory for Intelligent Systems and Controls
† Department of Mechanical and Aerospace Engineering, Cornell University, Ithaca, NY, USA
‡ Department of Mechanical Engineering and Materials Science, Duke University, Durham, NC, USA

Introduction and Motivation

This paper presents a multi-kernel probability distribution regression methodology that uses multi-layer reproducing kernel Hilbert space (RKHS) mappings to perform probability distribution to real and probability distribution to function regressions. The approach maps the distributions into RKHS by distribution embeddings. Then based on this RKHS, a multi-layer RKHS is constructed, within which the multi-kernel distribution regression can be implemented using an existing kernel regression algorithm, such as kernel recursive least squares (KRLS). The proposed algorithms are demonstrated through numerical simulations on synthetic data sets and compared with an existing algorithm. The results show that the proposed algorithm can outperform the exiting algorithm.

Formulation of Probability Distribution Regression Problems

- **Distribution to Real Regression (DRR)**:

$$\mathbf{z}_k = \mathcal{F}(P_k) + \boldsymbol{\epsilon}_k, \quad k = 1, \ldots, T \tag{1}$$

- **Distribution to Function Regression (DFR)**:

$$f_k = \mathcal{G}(P_k) + \boldsymbol{\varepsilon}_k, \quad k = 1, \ldots, T \tag{2}$$

where $P_k \in \mathcal{I}$ is a probability distribution defined in the probability space $\mathcal{I}$, $\mathbf{z}_k \in \mathbb{R}^{n_z}$ is the corresponding output response, and $\boldsymbol{\epsilon}_k$ is a zero mean Gaussian noise variable; $f_k \in \mathbb{F}$ is the function defined in a function space $\mathbb{F}$, $\boldsymbol{\varepsilon}_k$ is a zero mean Gaussian process.

DRR operator $\mathcal{F}: \mathcal{I} \mapsto \mathbb{R}^{n_z}$ **DFR operator** $\mathcal{G}: \mathcal{I} \mapsto \mathbb{F}$

Training Data sets:

- Distribution P_k is approximated by using data set $\mathcal{X}_k = \{\mathbf{x}_{k,1}, \ldots, \mathbf{x}_{k,s}, \ldots, \mathbf{x}_{k,N_k}\}$
- DDR operator $\mathcal{F}$ is approximated by using data set $\mathcal{D}_{\mathcal{F}} = \{(\mathcal{X}_1, \mathbf{z}_1), \ldots, (\mathcal{X}_T, \mathbf{z}_T)\}$
- Output function f_k is approximated by using data set $\mathcal{Y}_k = \{(\mathbf{y}_{k,1}, \mathbf{z}_{k,1}), \ldots, (\mathbf{y}_{k,M}, \mathbf{z}_{k,M})\}$ where $\mathbf{z}_{k,m} = f_k(\mathbf{y}_{k,m})$
- DFR operator $\mathcal{G}$ is approximated by using data set $\mathcal{D}_{\mathcal{G}} = \{(\mathcal{X}_1, \mathcal{Y}_1), \ldots, (\mathcal{X}_T, \mathcal{Y}_T)\}$.

Modified DFR operator:
According to (2)

$$\mathbf{z}_{k,m} = \mathcal{G}(P_k)(\mathbf{y}_{k,m}) + \boldsymbol{\varepsilon}_k(\mathbf{y}_{k,m}) \tag{3}$$

Define new DFR operator $\mathcal{G}'(P_k, \mathbf{y}_{k,m}) = \mathcal{G}(P_k)(\mathbf{y}_{k,m})$, then

$$\mathbf{z}_{k,m} = \mathcal{G}'(P_k, \mathbf{y}_{k,m}) + \boldsymbol{\epsilon}'_k \tag{4}$$

where the evaluation of $\boldsymbol{\varepsilon}(\mathbf{y}_{k,m})$ is denoted by $\boldsymbol{\epsilon}'_k$, which is a zero mean Gaussian noise.
Goal:

- Learn the operator $\mathcal{F}$ by using data set $\mathcal{D}_{\mathcal{F}}$
- Learn the operator $\mathcal{G}'$ by using data set $\mathcal{D}_{\mathcal{G}}$ instead of the operator $\mathcal{G}$

Multi-Kernel Distribution Regressions Methodology

Distribution Embeddings: Given a random variable (R.V.) $X \in \mathbb{R}^{n_x}$ associated with a distribution P_X and a corresponding Probability Density Function (PDF) p_X, an embedding $\boldsymbol{\mu}_X$ in RKHS can be defined as,

$$\boldsymbol{\mu}_X := \mathbf{E}_X[k_X(X, \cdot)] = \int p_X(\mathbf{x}) k_X(\mathbf{x}, \cdot) d(\mathbf{x}), \tag{5}$$

where $\mathbf{E}_X[\cdot]$ indicates the expectation operator, $k_X(\cdot, \cdot)$ is a kernel defined on $\mathbb{R}^{n_x} \times \mathbb{R}^{n_x}$ associated with RKHS $\mathcal{H}_X$. The distribution embedding $\boldsymbol{\mu}_X$ is also in the RKHS $\mathcal{H}_X$, provided $\mathbf{E}_X[k_X(X, X)] < \infty$. Its empirical estimate is

$$\hat{\boldsymbol{\mu}}_X = \frac{1}{N} \sum_{n=1}^{N} k_X(\mathbf{x}_n, \cdot) \tag{6}$$

where $\mathcal{D}_X = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ is a training set that is assumed to have been drawn *i.i.d* from P_X. With a characteristic kernel, the mapping from distribution P_X to the distribution embedding $\boldsymbol{\mu}_X \in \mathcal{H}_X$ is *injective*. A famous characteristic kernel is the *Gaussian kernel*, which is used in this paper to specify the kernel function $k_X(\cdot, \cdot)$.

Kernel Design and Multi-Layer RKHS

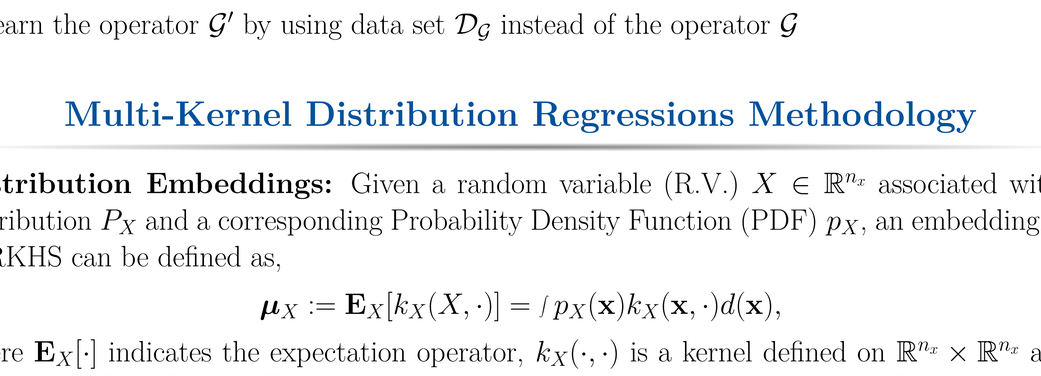

Figure 1: The frameworks of the multi-layer RKHS to implement MKDRR (red arrows) and MKDFR (blue arrows)

For DDR:

- Map the distribution data sets $\mathcal{X}_i$ and $\mathcal{X}_j$, $i, j = 1, \ldots, T$, into $\mathcal{H}_X$ as $\boldsymbol{\mu}_i$, $\boldsymbol{\mu}_j$, respectively.
- Define new kernel $\mathcal{K}_1$ and the corresponding RKHS $\mathcal{H}_1$,

$$\mathcal{K}_1(\boldsymbol{\mu}_i, \boldsymbol{\mu}_j) = \exp\left[-\frac{D(\boldsymbol{\mu}_i, \boldsymbol{\mu}_j)}{2\sigma_{\boldsymbol{\mu}}^2}\right] \tag{7}$$

  where $D(\boldsymbol{\mu}_i, \boldsymbol{\mu}_j) = \|\boldsymbol{\mu}_i - \boldsymbol{\mu}_j\|^2_{\mathcal{H}_X}$
- Approximate the mapping $\mathcal{F}$ using kernel methods in RKHS $\mathcal{H}_1$ with input signal $\boldsymbol{\mu}_k$ and desired signal $\mathbf{z}_k \in \mathbb{R}^{n_z}$, such as kernel adaptive filters.

For DFR:

- Map the distribution data sets $\mathcal{X}_i$ and into $\mathcal{H}_X$ as $\boldsymbol{\mu}_i$.
- Combine the distribution embedding $\boldsymbol{\mu}_i$ and $\mathbf{y}_{i,m}$ as a new term $\boldsymbol{\nu}_{i,m} = \left[\boldsymbol{\mu}_i^T, \mathbf{y}_{i,m}^T\right]^T \in \mathcal{H}_X \oplus \mathbb{R}^{n_y}$.
- Define new kernel $\mathcal{K}_2$ and the corresponding RKHS $\mathcal{H}_2$,

$$\mathcal{K}_2(\boldsymbol{\nu}_{i,m}, \boldsymbol{\nu}_{j,n}) = \exp\left[-\frac{(\boldsymbol{\nu}_{i,m} - \boldsymbol{\nu}_{j,n})^T \boldsymbol{\Sigma}_{XY}^{-1} (\boldsymbol{\nu}_{i,m} - \boldsymbol{\nu}_{j,n})}{2}\right] \tag{8}$$

- Approximate the mapping $\mathcal{G}$ using kernel methods in RKHS $\mathcal{H}_2$ with input signal $\boldsymbol{\nu}_{k,m}$ and desired signal $\mathbf{z}_{k,m} \in \mathbb{R}^{n_z}$, such as kernel adaptive filters.

Multi-Kernel Distribution to Real Regression based on KRLS

Multi-Kernel Distribution to Real Regression based on KRLS
Like the standard KRLS algorithm, the following cost function is minimized to learn the DRR defined in (1) from data sets $\mathcal{D}_{\mathcal{F}}$,

$$J_{DRR} = \min_{\boldsymbol{\omega}_{\mathcal{F}}} \left[\sum_{k=1}^{T} \left\|\mathbf{z}_k - \langle \boldsymbol{\omega}_{\mathcal{F}}, \mathcal{K}_{\mathcal{F}}(\boldsymbol{\mu}_k, \cdot)\rangle_{\mathcal{H}_{\mathcal{F}}}\right\|^2 + \lambda \left\|\boldsymbol{\omega}_{\mathcal{F}}\right\|^2_{\mathcal{H}_{\mathcal{F}}}\right] \tag{9}$$

where λ is a regularization factor. Then, the feature weight $\boldsymbol{\omega}_{\mathcal{F}}$ can be approximated at the kth iteration by

$$\boldsymbol{\omega}_{\mathcal{F}} = \boldsymbol{\Phi}_k \left[\boldsymbol{\Phi}_k^T \boldsymbol{\Phi}_k + \lambda \mathbf{I}_k\right]^{-1} \mathbf{Z}_k = \boldsymbol{\Phi}_k \mathbf{Q}_{\boldsymbol{\mu}}(k) \mathbf{Z}_k \tag{10}$$

where $\mathbf{I}_k$ is a $k \times k$ identity matrix, the feature matrices $\boldsymbol{\Phi}_k = [\mathcal{K}_{\mathcal{F}}(\boldsymbol{\mu}_1, \cdot), \ldots, \mathcal{K}_{\mathcal{F}}(\boldsymbol{\mu}_k, \cdot)]$ and desired matrices $\mathbf{Z}_k = [\mathbf{z}_1, \ldots, \mathbf{z}_k]^T$, and
Multi-Kernel Distribution to Function Regression based on KRLS Similarly, the following cost function is minimized to learn the DFR defined in (4) from data sets $\mathcal{D}_{\mathcal{G}}$,

$$J_{DFR} = \min_{\boldsymbol{\omega}_{\mathcal{G}}} \left[\sum_{k=1}^{T} \sum_{m=1}^{M} \left\|\mathbf{z}_{k,m} - \mathcal{F}_{\mathcal{G}}(P_k, \mathbf{y}_{k,m})\right\|^2 + \lambda \left\|\boldsymbol{\omega}_{\mathcal{G}}\right\|^2_{\mathcal{H}_{\mathcal{G}}}\right]. \tag{11}$$

we can approximate the feature weight $\boldsymbol{\omega}_{\mathcal{G}}$ at the kth by

$$\boldsymbol{\omega}_{\mathcal{G}} = \boldsymbol{\Upsilon}_k \left[\mathbf{K}_k + \lambda \mathbf{I}_{(kM)}\right]^{-1} \mathbf{V}_k = \boldsymbol{\Upsilon}_k \mathbf{Q}(k) \mathbf{V}_k, \tag{12}$$

where input feature matrices $\boldsymbol{\Psi}_k = [\mathcal{K}_{\mathcal{G}}(\boldsymbol{\nu}_{k,1}, \cdot), \ldots, \mathcal{K}_{\mathcal{G}}(\boldsymbol{\nu}_{k,M}, \cdot)]$ and $\boldsymbol{\Upsilon}_k = [\boldsymbol{\Psi}_1, \ldots, \boldsymbol{\Psi}_k]$, and the desired matrices $\mathbf{U}_k = [\mathbf{z}_{k,1}, \ldots, \mathbf{z}_{k,M}]^T$, $\mathbf{V}_k = \left[\mathbf{U}_k^T, \ldots, \mathbf{U}_k^T\right]^T$, and $\mathbf{K}_k = \boldsymbol{\Upsilon}_k^T \boldsymbol{\Upsilon}_k$.

Simulations and Results

Simulation settings

- the initial agent location distributions P_k, $k = 1, 2, \ldots$ on $\xi\eta$-coordinate plane., generated by 2D Mixture Gaussian distributions with two equivalently weighted components
- The means $[\mu_{\xi,i}, \mu_{\eta,i}]$, $i = 1, 2$, and covariance matrices $\boldsymbol{\Sigma}_i = \mathrm{diag}([\sigma_{\xi,i}, \sigma_{\eta,i}])$, $i = 1, 2$ are selected randomly,
- generate N_{train} training data sets, $N_{valid} = 25$ validation data sets, and $N_{test} = 50$ testing data sets.

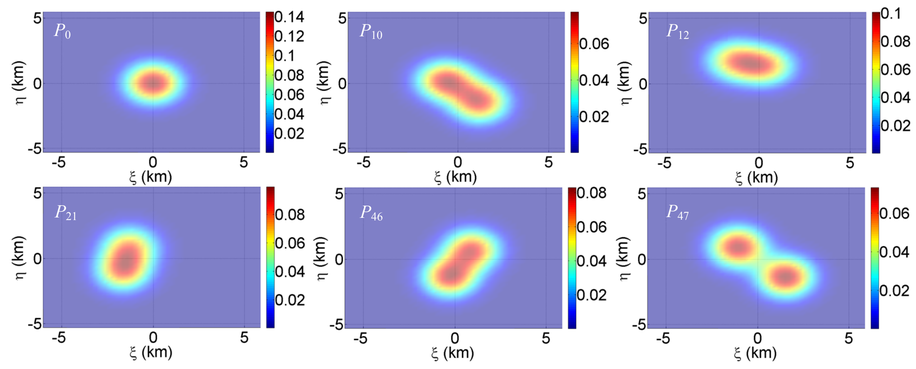

Figure 2: Examples of goal distribution and testing distributions generated by Gaussian mixture in DR problem.

Experiment of MKDDR-KRLS

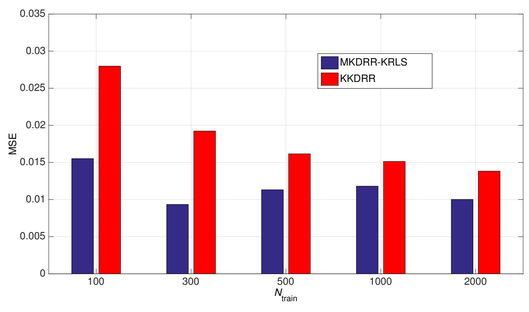

Figure 3: Performance comparison between MKDRR-KRLS and KKDRR algorithms

$$D_{CS}(P_k||P_0) = -\log \frac{\int p_k(\xi,\eta) p_0(\xi,\eta) d\xi d\eta}{\sqrt{\int p_k^2(\xi,\eta) d\xi d\eta \int p_0^2(\xi,\eta) d\xi d\eta}}, \tag{13}$$

Experiment of MKDFR-KRLS

Table 1: Regression performance results

Output functions	NMSE
CDF $F(\xi, \eta)$	0.0030 ± 0.0024
Gradient $g_\xi(\xi, \eta)$	0.0990 ± 0.0616
Gradient $g_\eta(\xi, \eta)$	0.0974 ± 0.0623

$$\mathrm{NMSE} = \frac{\sum_{m=1}^{M} \|\mathbf{z}_{k,m} - \hat{\mathbf{z}}_{k,m}\|^2}{\sum_{m=1}^{M} \|\mathbf{z}_{k,m}\|^2} \tag{14}$$

- Each sample set $\mathcal{X}_k$ has $N_{sample} = 500$ samples.
- $N_{train} = 500$ training distribution sets are used.

References and Acknowledgement

[1] Y. Engel, S. Mannor, and R. Meir, The kernel recursive least-squares algorithm, IEEE Transactions on Signal Processing, vol. 52, no. 8, pp. 2275 2285, Aug. 2004.
[2] W. Liu, P. Pokharel, and J. C. Principe,The Kernel Least Mean Square Algorithm, IEEE Transactions on Signal Processing, vol. 56, no. 2, pp. 543 554, Feb. 2009.
[3] B. Chen, S. Zhao, P. Zhu, and J. C. Principe, Quantized Kernel Least Mean Square Algorithm,âĂİ IEEE Transactions on Neural Networks and Learning Systems, vol. 23, no. 1, pp. 22 32, Jan. 2012.
[4] B. Chen, S. Zhao, P. Zhu, and J. C. Principe, Quantized Kernel Recursive Least Squares Algorithm, IEEE Transactions on Neural Networks and Learning Systems, vol. 24, no. 9, pp. 1484 1491, Sep. 2013
[5] L. Song, J. Huang, A. Smola, and K. Fukumizu, Hilbert Space Embeddings of Conditional Distributions with Applications to Dynamical Systems, In International Conference on Machine Learning (ICML), pp. 961âĂŞ968, 2009.
[6] P. Zhu, B. Chen, and J. C. Principe, Learning Nonlinear Generative Models of Time Series with a Kalman Filter in RKHS, IEEE Transactions on Signal Processing, vol. 62, no. 1, pp. 141 155, Jan. 2014.
[7] P. Zhu, Kalman Filtering in Reproducing Kernel Hilbert Spaces. Gainesivlle, FL, USA: PhD Thesis, Univeristy of Florida, 2013.
[8] K. Rudd, G. Foderaro, and S. Ferrari, A Generalized Reduced Gradient Method for the Optimal Control of Multiscale Dynamical Systems, Proc. of the IEEE Conference on Decision and Control (CDC), Dec. 2013.
[9] K. Kampa, E. Hasanbelliu, and J. C. Principe, Closed-form cauchyschwarz PDF divergence for mixture of Gaussians, International Joint Conference on Neural Networks (IJCNN), pp. 2578 2585, 2011.
This work was supported by NSF grant ECCS 1408022 and NFS grant DGE 1068871.