Monte Carlo Functional Regularisation for Continual Learning

Pengcheng Hao, Menghao Waiyan William Zhu, Ercan Engin Kuruoglu*
Institute of Data and Information, Tsinghua Shenzhen International Graduate School, Shenzhen, China
pengchenghao@sz.tsinghua.edu.cn, zhumh22@mails.tsinghua.edu.cn, kuruoglu@sz.tsinghua.edu.cn

***Abstract*—Continual learning (CL) is crucial for the adaptation of neural network models to new environments. Although outperforming weight-space regularisation approaches, the functional regularisation-based CL methods suffer from high computational costs and large linear approximation errors. In this work, we present a new functional regularisation CL framework, called MCFRCL, which approximates model prediction distributions by Monte Carlo (MC) sampling. Moreover, three continuous distributions are leveraged to capture the statistical characteristics of the MC samples via moment-based methods. Additionally, both the Wasserstein distance and the Kullback–Leibler (KL) distance are employed to construct the regularisation function. The proposed MCFRCL is evaluated against multiple benchmark methods on the MNIST and CIFAR datasets, with simulation results highlighting its effectiveness in both prediction accuracy and training efficiency.**

***Index Terms*—Continual learning, Functional regularisation, Monte Carlo sampling,**

I. Introduction

Continual learning (CL) [1], [2] refers to the ability of a neural network model to learn new tasks without forgetting previously acquired knowledge. This capability is especially valuable in dynamic environments where data evolves over time, eliminating the need to retrain models from scratch. For example, in healthcare, the CL supports the integration of new patient data into models without retraining, ensuring that the diagnostic systems stay current while safeguarding patient privacy [3]. Also, the CL is essential for autonomous driving systems to adapt to dynamic environments, such as different weather conditions [4]. Furthermore, a continual learning robot can continuously acquire new skills from unstructured real-world environments [5]. However, CL methods suffer from catastrophic forgetting [2], where a model tends to lose previously learned knowledge when adapting to new tasks.

To mitigate catastrophic forgetting, one presents weight-space regularisation CL methods, constraining the changes of model parameters during the acquisition of new knowledge. For instance, the Elastic Weight Consolidation (EWC) [6], [7] leverages the Fisher information matrix to guide weight regularisation. Also, the synaptic intelligence (SI) [8] constrains parameters according to their importance, which is evaluated via the entire training trajectory. Further, the CL method based on the Riemannian walk combines the regularization terms from SI and EWC, merging their respective advantages [9]. By contrast, the variational continual learning (VCL) [10] approximates the weight posterior distributions by variational inference (VI) [11], [12]. However, the complex relationship between model parameters and predictions makes the parameter-based regularisation ineffective in addressing catastrophic forgetting.

To address the limitations of weight regularisation, functional regularisation-based CL methods have been introduced, focusing on the intermediate or final output of neural networks. For instance, the functional regularisation for continual learning (FRCL) [13] combines inducing point Gaussian process (GP) [14] inference with deep neural networks but is limited to linear models. Also, the functional regularisation of the memorable past (FROMP) [15] method utilises the Laplace approximation [16] to estimate parameter variances, the optimisation of which is then not allowed. In contrast, the continual learning method via sequential function-Space variational inference (S-FSVI) [17] approximates prediction distributions by linearisation of a Bayesian neural network (BNN) [18], providing flexible optimisation over parameter means and variances. However, the S-FSVI requires high computational costs and large storage space due to the calculation of Jocabian matrices. Also, the adopted model linearisation introduces approximation errors.

In this work, we present a Monte Carlo (MC) sampling-based function-regularisation CL (MCFRCL) framework. The motivation for this choice is twofold. 1) The MC approach requires no Jacobian matrix computation and hence less computational loads. 2) BNNs are highly nonlinear systems, and compared with the linearisation method, the MC sampling can produce more precise uncertainty prediction. For instance, in nonlinear filtering, the classic extended Kalman filter [19] linearises the nonlinear system to estimate the prediction distribution. By contrast, the ensemble Kalman filter [20] employs MC samples and achieves better estimation results. The contributions of this work consist of:

1) To approximate the intractable model prediction distributions, we first obtain prediction samples from the current model and the previous task model by MC sampling;
2) The prediction distributions are then approximated by three continuous densities, including Gaussian, Laplace and Cauchy [21] distributions, the parameters of which

This work is supported by Tsinghua Shenzhen International Graduate School Start-up fund under Grant QD2022024C, Shenzhen Science and Technology Innovation Commission under Grant JCYJ20220530143002005 and Shenzhen Ubiquitous Data Enabling Key Lab under Grant ZDSYS20220527171406015. Corresponding author: Ercan Engin Kuruoglu.

are estimated by moment-based methods;
3) To construct a regularisation function, the Wasserstein [22] and Kullback–Leibler (KL) [23] distances are deployed to measure the similarity between the prediction distributions of the current and the previous models;
4) In the simulation, the proposed method is compared with various weight/function-space regularisation-based methods on the MNIST and CIFAR datasets.

The remainder of this paper is structured as follows: We begin, in Section II, with an introduction to the theoretical background. Subsequently, Section III describes our proposed MCFRCL method, and the simulation results are elucidated in Section IV. Besides, Section V concludes this study.

II. Theoretical preliminary

In this section, we introduce the employed 3 continuous distributions, followed by their corresponding moment-based parameter estimators and Wasserstein/KL distances.

TABLE I
Continuous distributions

Notations	Definitions
$\mathcal{N}(\mu, \sigma^2)$	Univariate Gaussian pdf with mean $\boldsymbol{\mu}$ and variance σ^2.
$\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$	Multivariate Gaussian pdf with mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$.
$\mathbf{Laplace}(a, b)$	Univariate Laplace pdf with location a and scale b.
$\mathbf{Cauchy}(l, \gamma)$	Univariate Cauchy pdf with location l and scale γ.

The definitions of the 3 densities are provided in Table I. Assume $x_{1:N} = \{x_\iota | \iota = 1, \ldots, N\}$ are N samples from an univariate distribution. For the Gaussian distribution, its parameters can be estimated by the mean and variance of the samples. Similarly, the parameters of $\mathbf{Laplace}(a, b)$ and $\mathbf{Cauchy}(l, \gamma)$ can be estimated as follows:

$$\begin{aligned} \hat{a} &= \mathrm{mean}\left(x_{1:N}\right), \hat{b} = \sqrt{\mathrm{var}\left(x_{1:N}\right)/2}, \\ \hat{l} &= \mathrm{median}\left(x_{1:N}\right), \hat{\gamma} = \mathrm{mad}\left(x_{1:N}\right), \end{aligned} \tag{1}$$

where the functions $\mathrm{mean}(.), \mathrm{var}(.), \mathrm{median}(.), \mathrm{mad}(.)$ represent the mean, variance, median, and median absolute deviation of the samples. Furthermore, the KL divergences between two univariate Gaussian [24], Laplace and Cauchy [21] distributions, denoted as $\mathbb{D}_{\mathrm{GKL}}$, $\mathbb{D}_{\mathrm{LKL}}$, $\mathbb{D}_{\mathrm{CKL}}$, can be computed as follows:

$$\begin{aligned} \mathbb{D}_{\mathrm{GKL}}\left(p_1||p_2\right) &= \log\frac{\sigma_2}{\sigma_1} + \frac{\sigma_1^2 + \left(\mu_1 - \mu_2\right)^2}{2\sigma_2^2} - \frac{1}{2} \\ \mathbb{D}_{\mathrm{LKL}}\left(p_1||p_2\right) &= \frac{b_1 \exp\left(-\frac{|a_1 - a_2|}{b_1}\right) + |a_1 - a_2|}{b_2} + \log\frac{b_2}{b_1} - 1 \\ \mathbb{D}_{\mathrm{CKL}}\left(p_1||p_2\right) &= \log\frac{(\gamma_1 + \gamma_2)^2 + (l_1 - l_2)^2}{4\gamma_1\gamma_2} \end{aligned} \tag{2}$$

where p_1, p_2 are two univariate densities and their corresponding parameters have the same subscripts. Besides, in multivariate case, the square of Wasserstein distance [25] between two Gaussian distributions, $\mathbb{D}_{\mathrm{GW}}$, can be written as

$$\begin{aligned} \mathbb{D}_{\mathrm{GW}}^2\left(p_1, p_2\right) &= ||\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2||_2^2 \\ &+ \mathrm{Tr}\left(\boldsymbol{\Sigma}_1 + \boldsymbol{\Sigma}_2 - 2\left(\boldsymbol{\Sigma}_1^{1/2}\boldsymbol{\Sigma}_2\boldsymbol{\Sigma}_1^{1/2}\right)^{1/2}\right). \end{aligned} \tag{3}$$

In comparison, the Laplace and Cauchy distributions do not have a closed-form Wasserstein distance.

III. MCFRCL

A. The proposed continual learning framework

Assume a series of sequentially arriving tasks with index $t \in \{1, \ldots, T\}$, where the dataset for the t-th task is denoted as $\mathcal{D}_t = (\mathbf{X}_t, \mathbf{y}_t)$. Considering a neural network $\mathbf{f} = f\left(.; \boldsymbol{\Theta}\right)$ with parameter $\boldsymbol{\Theta}$, the posterior estimation over the prediction function $\mathbf{f}$ can be expressed as

$$p\left(\mathbf{f}|\mathcal{D}_{1:t}\right) \propto p\left(\mathcal{D}_t|\mathbf{f}\right) p\left(\mathbf{f}|\mathcal{D}_{1:t-1}\right).$$

However, $p\left(\mathbf{f}|\mathcal{D}_{1:t}\right)$ is generally intractable. In [17], the sequential variational inference method is employed for the posterior approximation. Assume $q_t(\mathbf{f})$ is a functional variational distribution driven by a weight-space variational distribution $q_t(\boldsymbol{\Theta})$, the posterior approximation can be achieved by maximising

$$\mathcal{F} = \mathrm{E}_{q_t(\mathbf{f})}\left[\log p\left(\mathcal{D}_t|\mathbf{f}\right)\right] - \mathbb{D}_{\mathrm{KL}}\left[q_t\left(\mathbf{f}\right)||q_{t-1}\left(\mathbf{f}\right)\right],$$

where $\mathcal{F}$ is the evidence lower bound (ELBO) and $\mathbb{D}_{\mathrm{KL}}\left[q_t\left(\mathbf{f}\right)||q_{t-1}\left(\mathbf{f}\right)\right]$ is the function-space regularisation term. Since there is no closed-form solution to $\mathcal{F}$, an approximation is proposed in [17]:

$$\begin{aligned} \mathcal{F} \approx &\frac{1}{S_\beta}\sum_{i=1}^{S_\beta} \log p\left(y_\beta|\mathbf{f}_i^\beta\right) \\ &- \sum_{\tau=1}^{t-1}\sum_{k=1}^{D_\tau} \mathbb{D}_{\mathrm{KL}}\left[q_t\left(\mathbf{f}_k^{\mathcal{C}_\tau}\right)||q_{t-1}\left(\mathbf{f}_k^{\mathcal{C}_\tau}\right)\right] \end{aligned} \tag{4}$$

where the prediction samples $\mathbf{f}_i^\beta = f\left(\mathbf{X}_\beta; \boldsymbol{\Theta}_i\right)$, $\mathbf{X}_\beta \in \mathbf{X}_t$ represents a batch of training data, $\boldsymbol{\Theta}_i$ is the i-th sample of $q_t(\boldsymbol{\Theta})$ and S_β is the number of the prediction samples. Also, the number of model output dimensions for the τ-th task is D_τ. Besides, the output of the k-th dimension is $\mathbf{f}_k^{\mathcal{C}_\tau} = \left[f\left(\mathbf{X}_\tau^{\mathcal{C}}; \boldsymbol{\Theta}\right)\right]_k$, where the context set $\mathbf{X}_\tau^{\mathcal{C}}$ consists of $N_{\mathcal{C}_\tau}$ representative samples in the coreset of the τ-th task and the coreset is sampled from the corresponding training dataset.

Considering the benefits of both the KL and Wasserstein distance in functional regularisation [17], [26], we present our new objective function as

$$\begin{aligned} \mathcal{F} \approx &\frac{1}{S_\beta}\sum_{i=1}^{S_\beta} \log p\left(y_\beta|\mathbf{f}_i^\beta\right) \\ &- \lambda\frac{N_\beta}{N_{\mathcal{C}_\tau}}\sum_{\tau=1}^{t-1}\sum_{k=1}^{D_\tau} \mathbb{D}\left[q_t\left(\mathbf{f}_k^{\mathcal{C}_\tau}\right)||q_{t-1}\left(\mathbf{f}_k^{\mathcal{C}_\tau}\right)\right], \end{aligned} \tag{5}$$

where $\mathbb{D} \in \{\mathbb{D}_{\mathrm{GKL}}, \mathbb{D}_{\mathrm{LKL}}, \mathbb{D}_{\mathrm{CKL}}, \mathbb{D}_{\mathrm{GW}}\}$. Also, N_β is the batch size, $\frac{N_\beta}{N_{\mathcal{C}_\tau}}$ is used to alleviate the influence of the unbalanced data points for different tasks, and λ is a scalar regularisation coefficient. However, the functional regularisation term in equation (5) is intractable, as there is no closed-form solution to the prediction distributions $q_t\left(\mathbf{f}_k^{\mathcal{C}_\tau}\right)$ and $q_{t-1}\left(\mathbf{f}_k^{\mathcal{C}_\tau}\right)$.

B. Approximation of the regularisation function

To handle the intractable prediction distribution, this section employs an MC sampling-based method to approximate the two variational distributions and then present an estimator for $\mathbb{D}\left[q_t\left(\mathbf{f}_k^{\mathcal{C}_\tau}\right)||q_{t-1}\left(\mathbf{f}_k^{\mathcal{C}_\tau}\right)\right]$ as shown in Figure 1. There are three steps:

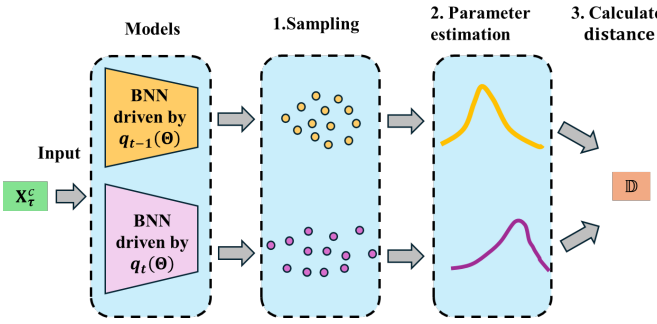

Fig. 1. The flowchart of the approximation method of the MCFRCL functional regularisation.

1) Monte Carlo sampling: With $\mathbf{X}_\tau^{\mathcal{C}}, \tau = 1, \cdots, t-1$, we draw samples $\left[\mathbf{f}_k^{\mathcal{C}_\tau^+}\right]_j$ and $\left[\mathbf{f}_k^{\mathcal{C}_\tau^-}\right]_j$, $j = 1, \ldots, S_{\mathcal{C}}$, from $q_t\left(\mathbf{f}_k^{\mathcal{C}_\tau}\right)$ and $q_{t-1}\left(\mathbf{f}_k^{\mathcal{C}_\tau}\right)$, respectively, where $S_{\mathcal{C}}$ is the number of samples. Also, $\left[\mathbf{f}_k^{\mathcal{C}_\tau^+}\right]_j = f\left(\mathbf{X}; \mathbf{\Theta}_j^+\right)$, $\left[\mathbf{f}_k^{\mathcal{C}_\tau^-}\right]_j = f\left(\mathbf{X}; \mathbf{\Theta}_j^-\right)$ where $\Theta_j^+ \sim q_t(\mathbf{\Theta})$ and $\Theta_j^- \sim q_{t-1}(\mathbf{\Theta})$.

2) Parameter estimation of the prediction distribution: We approximate $q_t\left(\mathbf{f}_k^{\mathcal{C}_\tau}\right)$ and $q_{t-1}\left(\mathbf{f}_k^{\mathcal{C}_\tau}\right)$ by the continuous densities in Table I. To reduce the computational cost, we assume that all the components of $\mathbf{f}_k^{\mathcal{C}_\tau}$ are mutually independent, and hence the high-dimensional functional distribution approximation problem can be transformed into multiple one-dimensional estimation tasks. Given $\mathbf{f}_{k,\xi}^{\mathcal{C}_\tau}$, $\xi = 1, \ldots, N_{\mathcal{C}_\tau}$, is the ξ-th element of $\mathbf{f}_k^{\mathcal{C}_\tau}$, we have its corresponding one-dimensional samples $\left[\mathbf{f}_{k,\xi}^{\mathcal{C}_\tau^+}\right]_j$ and $\left[\mathbf{f}_{k,\xi}^{\mathcal{C}_\tau^-}\right]_j$. Then we obtain the approximated univariate distributions $\hat{q}_t\left(\mathbf{f}_{k,\xi}^{\mathcal{C}_\tau}\right)$ and $\hat{q}_{t-1}\left(\mathbf{f}_{k,\xi}^{\mathcal{C}_\tau}\right)$, of which parameters can be estimated by (1).

Remark 1: The adopted moment-based estimators in (1) are computationally efficient and differentiable, which is beneficial to the model training process.

3) Calculation of the distance $\mathbb{D}$: For $\mathbb{D} \in \{\mathbb{D}_{\mathrm{GKL}}, \mathbb{D}_{\mathrm{LKL}}, \mathbb{D}_{\mathrm{CKL}}\}$, as the KL distance is additive for independent distributions, we have

$$\mathbb{D}\left[q_t\left(\mathbf{f}_k^{\mathcal{C}_\tau}\right)||q_{t-1}\left(\mathbf{f}_k^{\mathcal{C}_\tau}\right)\right] \approx \sum_{\xi=1}^{N_{\mathcal{C}_\tau}} \mathbb{D}\left[\hat{q}_t\left(\mathbf{f}_{k,\xi}^{\mathcal{C}_\tau}\right)||\hat{q}_{t-1}\left(\mathbf{f}_{k,\xi}^{\mathcal{C}_\tau}\right)\right].$$

By contrast, for $\mathbb{D}_{\mathrm{GW}}$, we have

$$\mathbb{D}_{\mathrm{GW}}^2\left[q_t\left(\mathbf{f}_k^{\mathcal{C}_\tau}\right)||q_{t-1}\left(\mathbf{f}_k^{\mathcal{C}_\tau}\right)\right] \approx \sum_{\xi=1}^{N_{\mathcal{C}_\tau}} \mathbb{D}_{\mathrm{GW}}^2\left[\hat{q}_t\left(\mathbf{f}_{k,\xi}^{\mathcal{C}_\tau}\right)||\hat{q}_{t-1}\left(\mathbf{f}_{k,\xi}^{\mathcal{C}_\tau}\right)\right], \tag{6}$$

which can be easily derived from (3).

C. Discussion

Our proposed method is mostly relevant to the function-space regularisation methods, including FRCL [13], FROMP [15], and S-FSVI [17]. The FRCL only treats the weights of the last layer in a neural network as random, whilst our method is applicable to fully stochastic models. Also, compared with the FROMP, the MCFRCL achieves an optimisation on both means and variances of parameters. Furthermore, unlike the S-FSVI, which relies on linear approximations, the MCFRCL utilises MC sampling. This approach avoids the expensive computations of Jocabian matrices and generally provides more accurate predictions, especially in highly nonlinear systems.

IV. Empirical Evaluation

In this section, we evaluate our proposed MCFRCL method. Section IV-A and IV-B introduce the CL tasks based on the (Fashion) MNIST and CIFAR datasets, respectively. In Section IV-C and IV-D, various CL approaches are compared with four MCFRCL variants based on $\{\mathbb{D}_{\mathrm{GKL}}, \mathbb{D}_{\mathrm{LKL}}, \mathbb{D}_{\mathrm{CKL}}, \mathbb{D}_{\mathrm{GW}}\}$, with benchmark results directly sourced from [17] and [27] to ensure strong baselines.

A. Split (Fashion) MNIST setup

1) Split (Fashion) MNIST comprises five tasks, each involving binary classification between a pair of (Fashion) MNIST classes. Also, both the MNIST and Fashion MNIST datasets contain 60,000 samples for training and 10,000 samples for testing, and all images are transformed into floating-point numbers ranging from 0 to 1. 2) We employ single-head fully connected neural networks with two hidden layers of size 256, applying the ReLU activation function to all non-output units. Besides, an Adam optimiser with an initial learning rate of 0.0005 ($\beta_1 = 0.9; \beta_2 = 0.999$) is adopted. 3) For the split MNIST (S-MNIST) experiment, we evaluate two scenarios with 40 and 200 coreset points per task, using 10 and 60 epochs, respectively. By contrast, for the split Fashion MNIST (S-FMNIST) tasks, the coreset size is manually set to 200 points per task, with 5 epochs per task. Additionally, during training on the first task, context points are generated by

sampling each pixel uniformly from the range $[0, 1]$. For subsequent tasks, context points are randomly selected from the coreset. 4) For the first task, we assume a Gaussian functional prior distribution with zero mean and a diagonal covariance of magnitude 0.001. 5) Set N_β=128, $N_{\mathcal{C}_\tau}$=40, S_β=30, $S_\mathcal{C}$=30. Also, λ is selected from $\{10^n, 3 \times 10^n \mid n \in [-9, 7], n \in \mathbb{Z}\}$, and we set the optimal result as the final result. Besides, all (Fashion) MNIST experiments are conducted with 10 MC runs.

B. Split CIFAR setup

1) Split CIFAR [15] comprises six tasks. The first involves ten-way classification using the entire CIFAR-10 dataset, while each of the remaining five tasks also involves ten-way classification with classes selected from CIFAR-100. 2) As in [17], we utilize a neural network consisting of four convolutional layers, followed by two fully connected layers and multiple output heads, one for each task. Also, we use the same Adam optimiser as in the Split (Fashion) MNIST setup. 3) The coreset size is fixed at 200, with 120 epochs for the first task and 50 epochs for the subsequent tasks. Also, for the first task, context points are generated by sampling each pixel uniformly from the range $[0, 1]$, while context points are randomly selected from the coreset in the subsequent tasks. 4) For the first task, the functional prior distribution is assumed to be Gaussian with zero mean and a diagonal covariance of magnitude 1.0. 5) Set N_β=512, $N_{\mathcal{C}_\tau}$=50, S_β=5, $S_\mathcal{C}$=30. Also, λ is selected from $\{10^n \mid n \in [-9, 2], n \in \mathbb{Z}\}$, and we set the optimal result as the final result. Besides, 5 MC runs are given for all CIFAR experiments.

C. Performance on the MNIST dataset

In this experiment, the MCFRCL variants are compared with various benchmark methods, including Online EWC [7], SI [8], VCL [10], VAR-GP [28], FROMP [15], S-FSVI [17]. The prediction accuracy comparison is shown in Table II, and the bold numbers highlight the best results. The optimal MCFRCL consistently outperforms the other benchmark methods across all scenarios. Also, the optimal MCFRCL variant varies depending on the datasets, while the MCFRCL with $\mathbb{D}_{\mathrm{CKL}}$ produces worse estimation than the other variants. This indicates that the heavy-tailed Cauchy distribution fails to accurately capture the prediction uncertainty.

Also, Figure 2 presents the average training time per epoch and required GPU memory of the optimal MCFRCL variant and the S-FSVI. In all 3 scenarios, the MCFRCL requires less training time and GPU memory as the S-FSVI suffers from the computationally expensive Jacobian matrix.

Besides, the influence of the MC sample sizes $S_\mathcal{C}$ and S_β are illustrated in Table III. Due to the computational cost, our method is not scalable to large sample sizes. From Table III, the small changes in sample sizes have little influence on model performance.

D. Performance on the CIFAR dataset

The performance of the MCFRCL on the CIFAR dataset is presented in Table IV. There are 2 scores, the prediction accuracy and backward transfer (BT), and higher values are better for both. The bold numbers indicate the best results. Our proposed method produces more accurate predictions than the weight-regularisation CL methods, including the EWC and the VCL. However, compared with the functional regularisation-based benchmarks, the MCFRCL performs worse, which suggests the limitations of the MC sampling in capturing complex statistical characteristics of large stochastic models. Also, the MCFRCL has lower BT scores than the other methods, which implies that it suffers serious catastrophic forgetting in this scenario.

TABLE II
COMPARISON OF PREDICTION ACCURACY (%) ON MNISTS

Method	S-MNIST 40 pts/task	S-MNIST 200 pts/task	S-FMNIST 200 pts/task
Online EWC	19.95±0.28	19.95±0.28	19.95±0.28
SI	19.82±0.09	19.82±0.09	19.80±0.21
VCL	22.31±2.00	32.11±1.16	53.59±3.74
VAR-GP	-	90.57±1.06	-
FROMP	75.21±2.05	89.54±0.72	78.83±0.46
S-FSVI	84.51±1.30	92.87±0.14	77.54±0.40
MCFRCL:			
$\mathbb{D}_{\mathrm{GW}}$	83.30±1.82	**93.22±0.39**	78.3±4.33
$\mathbb{D}_{\mathrm{GKL}}$	82.43±1.31	92.63±0.44	**79.15±1.02**
$\mathbb{D}_{\mathrm{LKL}}$	**84.85± 0.88**	92.88±0.42	73.76±6.23
$\mathbb{D}_{\mathrm{CKL}}$	65.84±5.77	89.59±1.51	33.14±5.67

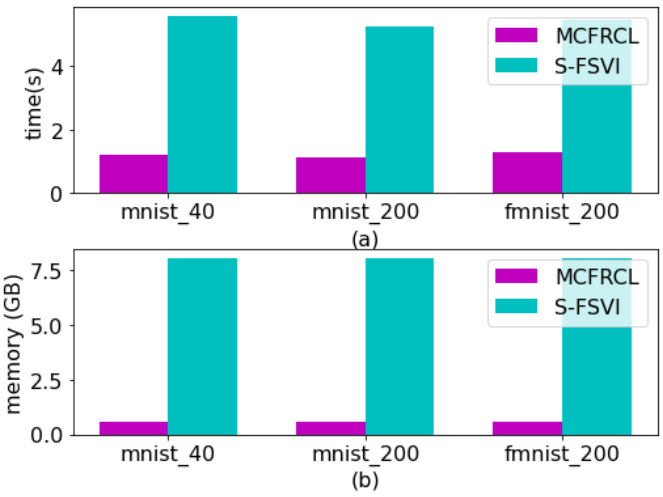

Fig. 2. Average training time per epoch and occupied GPU memory during the last task.

V. CONCLUSIONS

In this work, we introduce the MCFRCL, a novel functional regularisation-based continual learning framework, where three continuous distributions approximate the model prediction distributions via MC sampling and moment-based methods. Also, both the Wasserstein and KL distances are deployed to construct the regularisation function. Our proposed method is compared with various benchmark CL frameworks. The experiment results demonstrate that the MCFRCL achieves better prediction accuracy and training efficiency on MNIST

TABLE III
COMPARISON OF PREDICTION ACCURACY (%) WITH DIFFERENT NUMBERS OF MC SAMPLES ON FMNIST

		$S_{\mathcal{C}}$=5	$S_{\mathcal{C}}$=30	$S_{\mathcal{C}}$=100
$\mathbb{D}_{\mathrm{GW}}$	S_β=10	79.02±3.81	77.50±4.68	78.08±4.12
	S_β=30	79.96±3.00	**80.12±2.95**	78.54±4.55
	S_β=100	78.21±4.44	77.33±4.60	79.01±3.83
$\mathbb{D}_{\mathrm{GKL}}$	S_β=10	79.25±2.67	78.33±3.05	78.05±2.85
	S_β=30	79.22±2.77	**79.33±0.70**	78.35±2.78
	S_β=100	78.95±2.90	78.30±2.90	78.01±3.05

TABLE IV
PERFORMANCE COMPARISON ON CIFAR DATASETS

Method	Accuracy(%)	BT
EWC	71.6±0.4	**-2.3±0.6**
VCL	67.4±0.6	-9.2±0.8
FROMP	76.2±0.2	-2.6±0.4
S-FSVI	**77.6±0.2**	-2.5±0.2
MCFRCL:		
$\mathbb{D}_{\mathrm{GW}}$	73.08±0.2	-7.99±0.4
$\mathbb{D}_{\mathrm{GKL}}$	72.49±0.2	-8.29±0.2
$\mathbb{D}_{\mathrm{LKL}}$	73.16±1.1	-8.43±0.7
$\mathbb{D}_{\mathrm{CKL}}$	67.8±0.5	-14.47±0.5

datasets. By contrast, on the more complicated CIFAR dataset, while outperforming the weigh-regularisation methods, the MCFRCL falls short compared to other function-regularisation benchmarks. This suggests that the employed MC sampling is ineffective at approximating complex model prediction densities with a small number of samples. In the future, we will consider applying our proposed method to some edge devices, such as health monitoring systems [29], which require light and fast models due to the limited computational power.

REFERENCES

[1] L. Wang, X. Zhang, H. Su, and J. Zhu, "A comprehensive survey of continual learning: theory, method and application," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

[2] M. De Lange, R. Aljundi, M. Masana, S. Parisot, X. Jia, A. Leonardis, G. Slabaugh, and T. Tuytelaars, "A continual learning survey: Defying forgetting in classification tasks," *IEEE transactions on pattern analysis and machine intelligence*, vol. 44, no. 7, pp. 3366–3385, 2021.

[3] T. Verma, L. Jin, J. Zhou, J. Huang, M. Tan, B. C. M. Choong, T. F. Tan, F. Gao, X. Xu, D. S. Ting *et al.*, "Privacy-preserving continual learning methods for medical image classification: a comparative analysis," *Frontiers in Medicine*, vol. 10, p. 1227515, 2023.

[4] E. Verwimp, K. Yang, S. Parisot, L. Hong, S. McDonagh, E. Pérez-Pellitero, M. De Lange, and T. Tuytelaars, "Clad: A realistic continual learning benchmark for autonomous driving," *Neural Networks*, vol. 161, pp. 659–669, 2023.

[5] S. Auddy, J. Hollenstein, M. Saveriano, A. Rodríguez-Sánchez, and J. Piater, "Continual learning from demonstration of robotics skills," *Robotics and Autonomous Systems*, vol. 165, p. 104427, 2023.

[6] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska *et al.*, "Overcoming catastrophic forgetting in neural networks," *Proceedings of the national academy of sciences*, vol. 114, no. 13, pp. 3521–3526, 2017.

[7] J. Schwarz, W. Czarnecki, J. Luketina, A. Grabska-Barwinska, Y. W. Teh, R. Pascanu, and R. Hadsell, "Progress & compress: A scalable framework for continual learning," in *International conference on machine learning*. PMLR, 2018, pp. 4528–4537.

[8] F. Zenke, B. Poole, and S. Ganguli, "Continual learning through synaptic intelligence," in *International conference on machine learning*. PMLR, 2017, pp. 3987–3995.

[9] A. Chaudhry, P. K. Dokania, T. Ajanthan, and P. H. Torr, "Riemannian walk for incremental learning: Understanding forgetting and intransigence," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 532–547.

[10] C. V. Nguyen, Y. Li, T. D. Bui, and R. E. Turner, "Variational continual learning," in *International Conference on Learning Representations*, 2018. [Online]. Available: https://openreview.net/forum?id=BkQqq0gRb

[11] D. M. Blei, A. Kucukelbir, and J. D. McAuliffe, "Variational inference: A review for statisticians," *Journal of the American statistical Association*, vol. 112, no. 518, pp. 859–877, 2017.

[12] C. Zhang, J. Bütepage, H. Kjellström, and S. Mandt, "Advances in variational inference," *IEEE transactions on pattern analysis and machine intelligence*, vol. 41, no. 8, pp. 2008–2026, 2018.

[13] M. K. Titsias, J. Schwarz, A. G. de G. Matthews, R. Pascanu, and Y. W. Teh, "Functional regularisation for continual learning with gaussian processes," in *International Conference on Learning Representations*, 2020. [Online]. Available: https://openreview.net/forum?id=HkxCzeHFDB

[14] E. Schulz, M. Speekenbrink, and A. Krause, "A tutorial on gaussian process regression: Modelling, exploring, and exploiting functions," *Journal of mathematical psychology*, vol. 85, pp. 1–16, 2018.

[15] P. Pan, S. Swaroop, A. Immer, R. Eschenhagen, R. Turner, and M. E. E. Khan, "Continual deep learning by functional regularisation of memorable past," *Advances in neural information processing systems*, vol. 33, pp. 4453–4464, 2020.

[16] A. Kristiadi, M. Hein, and P. Hennig, "Learnable uncertainty under laplace approximations," in *Uncertainty in Artificial Intelligence*. PMLR, 2021, pp. 344–353.

[17] T. G. Rudner, F. B. Smith, Q. Feng, Y. W. Teh, and Y. Gal, "Continual learning via sequential function-space variational inference," in *International Conference on Machine Learning*. PMLR, 2022, pp. 18 871–18 887.

[18] L. V. Jospin, H. Laga, F. Boussaid, W. Buntine, and M. Bennamoun, "Hands-on bayesian neural networks—a tutorial for deep learning users," *IEEE Computational Intelligence Magazine*, vol. 17, no. 2, pp. 29–48, 2022.

[19] S. Julier and J. K. Uhlmann, "A general method for approximating nonlinear transformations of probability distributions," *Citeseer*, 1996.

[20] G. Evensen, "Sequential data assimilation with a nonlinear quasi-geostrophic model using monte carlo methods to forecast error statistic," *JOURNAL OF GEOPHYSICAL RESEARCH*, vol. 99, no. C5, pp. 10,143–10,162, 1994.

[21] F. Chyzak and F. Nielsen, "A closed-form formula for the kullback-leibler divergence between cauchy distributions," *arXiv preprint arXiv:1905.10965*, 2019.

[22] V. M. Panaretos and Y. Zemel, "Statistical aspects of wasserstein distances," *Annual review of statistics and its application*, vol. 6, no. 1, pp. 405–431, 2019.

[23] T. Van Erven and P. Harremos, "Rényi divergence and kullback-leibler divergence," *IEEE Transactions on Information Theory*, vol. 60, no. 7, pp. 3797–3820, 2014.

[24] Y. Zhang, J. Pan, L. K. Li, W. Liu, Z. Chen, X. Liu, and J. Wang, "On the properties of kullback-leibler divergence between multivariate gaussian distributions," *Advances in Neural Information Processing Systems*, vol. 36, 2024.

[25] C. R. Givens and R. M. Shortt, "A class of wasserstein metrics for probability distributions." *Michigan Mathematical Journal*, vol. 31, no. 2, pp. 231–240, 1984.

[26] V. D. Wild, R. Hu, and D. Sejdinovic, "Generalized variational inference in function spaces: Gaussian measures meet bayesian deep learning," *Advances in Neural Information Processing Systems*, vol. 35, pp. 3716–3730, 2022.

[27] A. Scannell, R. Mereu, P. E. Chang, E. Tamir, J. Pajarinen, and A. Solin, "Function-space parameterization of neural networks for sequential learning," in *The Twelfth International Conference on Learning Representations*, 2024. [Online]. Available: https://openreview.net/forum?id=2dhxxIKhqz

[28] S. Kapoor, T. Karaletsos, and T. D. Bui, "Variational auto-regressive gaussian processes for continual learning," in *International Conference on Machine Learning*. PMLR, 2021, pp. 5290–5300.

[29] G. Sivapalan, K. K. Nundy, S. Dev, B. Cardiff, and D. John, "Annet: A lightweight neural network for ecg anomaly detection in iot edge sensors," *IEEE Transactions on Biomedical Circuits and Systems*, vol. 16, no. 1, pp. 24–35, 2022.