Theory on Mixture-of-Experts in Continual Learning

Hongbo Li[1,3], **Sen Lin**[2], **Lingjie Duan**[1], **Yingbin Liang**[3], **Ness Shroff**[3,4]

[1]Engineering Systems and Design Pillar, Singapore University of Technology and Design,
[2]Department of Computer Science, University of Houston,
[3]Department of Electrical and Computer Engineering, The Ohio State University
[4]Department of Computer Science and Engineering, The Ohio State University

MOTIVATIONS

- Continual Learning (CL) has emerged as an important paradigm in machine learning, in which an expert aims to learn a sequence of tasks one by one over time.
- Given the dynamic nature of CL, one major challenge herein is known as catastrophic forgetting, where a single expert can perform poorly on (i.e., easily forget) the previous tasks when learning new tasks if data distributions change largely across tasks.

Literature Review: CL

Various empirical approaches have been proposed to tackle catastrophic forgetting in CL:

- Regularization-based approaches (e.g., Kirkpatrick et al. 2017; Gou et al. 2021).
- Parameter-isolation-based approaches (e.g., Chaudhry et al. 2018; Konishi et al. 2023).
- Memory-based approaches (e.g., Jin et al. 2021; S. Lin, Yang, et al. 2021; Gao and Liu 2023).

On the other hand, theoretical studies on CL are very limited.

Literature Review: MoE Model

- Mixture-of-Experts (MoE) has found widespread applications in emerging fields such as large language models (LLMs) (e.g., Du et al. 2022; Li et al. 2024; B. Lin et al. 2024).
- Chen et al. (2022) theoretically analyze the mechanism of MoE in deep learning under the setup of a mixture of classification problem. However, this study focuses on a single-task setting, and hence does not analyze the dynamics of CL.

Literature Review: MoE in CL

- Recently, the MoE model has been applied to reducing catastrophic forgetting in CL (Hihn and Braun 2021; Wang et al. 2022; Doan, Mirzadeh, and Farajtabar 2023; Rypeść et al. 2023; J. Yu et al. 2024).
- However, these works solely focus on empirical methods, lacking theoretical analysis of how the MoE performs in CL.

CL in Linear Model

We consider the CL setting with T training rounds.

- In each round $t \in [T]$, one out of N tasks randomly arrives to be learned by the MoE model with M experts.
- For each task, we consider fitting a linear model $f(\mathbf{X}) = \mathbf{X}^\top \boldsymbol{w}$ with ground truth $\boldsymbol{w} \in \mathbb{R}^d$.
- Then for the t-th task arrival, let $\mathcal{D}_t = (\mathbf{X}_t, \mathbf{y}_t)$ denote its dataset, where $\mathbf{X}_t \in \mathbb{R}^{d \times s_t}$ is the feature matrix, and $\mathbf{y}_t \in \mathbb{R}^{s_t}$ is the output vector.
- In this study, we focus on the overparameterized regime, where $s_t < d$.

CL in Linear Model (cont.)

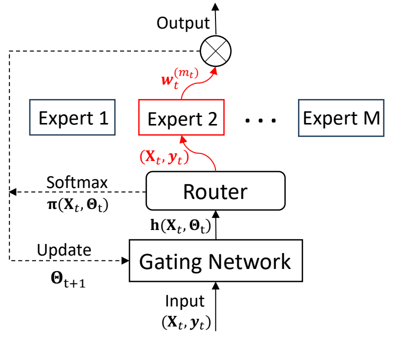
- Let $\mathcal{W} = \{\boldsymbol{w}_1, \cdots, \boldsymbol{w}_N\}$ represent the collection of ground truth vectors of all N tasks.
- For any two tasks $n, n' \in [N]$, we assume $\|\boldsymbol{w}_n - \boldsymbol{w}_{n'}\|_\infty = \mathcal{O}(\sigma_0)$, where $\sigma_0 \in (0, 1)$ denotes the variance.
- We assume that task n possesses a unique feature signal $\boldsymbol{v}_n \in \mathbb{R}^d$ with $\|\boldsymbol{v}_n\|_\infty = \mathcal{O}(1)$.
- In each round $t \in [T]$, let $n_t \in [N]$ denote the index of the current task arrival with ground truth $\boldsymbol{w}_{n_t} \in \mathcal{W}$.

CL IN LINEAR MODEL (CONT.)

At the beginning of each training round $t \in [T]$, the dataset $\mathcal{D}_t = (\mathbf{X}_t, \mathbf{y}_t)$ of the new task arrival n_t is generated by the following steps:

1. Uniformly draw a ground truth $\boldsymbol{w}_n$ from ground-truth pool $\mathcal{W}$ and let $\boldsymbol{w}_{n_t} = \boldsymbol{w}_n$.
2. Independently generate a random variable $\beta_t \in (0, C]$, where C is a constant satisfying $C = \mathcal{O}(1)$.
3. Generate $\mathbf{X}_t$ as a collection of s_t samples, where one sample is given by $\beta_t \boldsymbol{v}_{n_t}$ and the rest of the $s_t - 1$ samples are drawn from normal distribution $\mathcal{N}(\mathbf{0}, \sigma_t^2 \boldsymbol{I}_d)$, where $\sigma_t \geq 0$ is the noise level.
4. Generate the output to be $\mathbf{y}_t = \mathbf{X}_t^\top \boldsymbol{w}_{n_t}$.

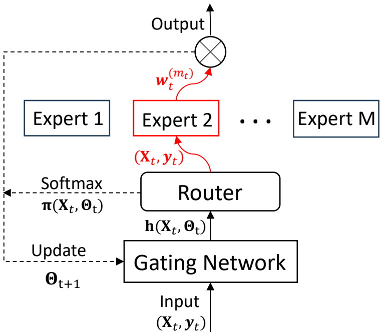
Structure of the MoE Model



- Upon the arrival of task n_t and input of its data $\mathcal{D}_t = (\mathbf{X}_t, \mathbf{y}_t)$, the gating network computes its linear output $h_m(\mathbf{X}_t, \boldsymbol{\theta}_t^{(m)})$ for each expert $m \in [M]$.
- Define $\mathbf{h}(\mathbf{X}_t, \boldsymbol{\Theta}_t) := [h_1(\mathbf{X}_t, \boldsymbol{\theta}_t^{(1)}) \ \cdots \ h_M(\mathbf{X}_t, \boldsymbol{\theta}_t^{(M)})]$ and $\boldsymbol{\Theta}_t := [\boldsymbol{\theta}_t^{(1)} \ \cdots \ \boldsymbol{\theta}_t^{(M)}]$ as the outputs and the parameters of the gating network for all experts, respectively. We obtain

$$\mathbf{h}(\mathbf{X}_t, \boldsymbol{\Theta}_t) = \sum_{i \in [s_t]} \boldsymbol{\Theta}_t^\top \mathbf{X}_{t,i}$$

Structure of the MoE Model (cont.)



- In each round t, for task n_t, the router selects the expert with the maximum gate output $h_m(\mathbf{X}_t, \boldsymbol{\theta}_t^{(m)})$, denoted as m_t, from the M experts:

$$m_t = \arg\max_m \{h_m(\mathbf{X}_t, \boldsymbol{\theta}_t^{(m)}) + r_t^{(m)}\},$$

  where $r_t^{(m)}$ for any $m \in [M]$ is drawn independently from the uniform distribution $\mathrm{Unif}[0, \lambda]$.
- Additionally, the router calculates the softmaxed gate outputs, derived by

$$\pi_m(\mathbf{X}_t, \boldsymbol{\Theta}_t) = \frac{\exp(h_m(\mathbf{X}_t, \boldsymbol{\theta}_t^{(m)}))}{\sum_{m'=1}^{M} \exp(h_{m'}(\mathbf{X}_t, \boldsymbol{\theta}_t^{(m)}))}, \quad \forall m \in [M]$$

  for updating $\boldsymbol{\Theta}_{t+1}$.

Training of the Expert Model

- Let $\boldsymbol{w}_t^{(m)}$ denote the model of expert m in the t-th training round, where each model is initialized from zero.
- In each round t, the training loss is defined by the mean-squared error (MSE) relative to $\mathcal{D}_t$:

$$\mathcal{L}_t^{tr}(\boldsymbol{w}_t^{(m_t)}, \mathcal{D}_t) = \frac{1}{s_t}\|(\mathbf{X}_t)^\top \boldsymbol{w}_t^{(m_t)} - \mathbf{y}_t\|_2^2.$$

Training of the Expert Model (cont.)

- Gradient descent (GD) provides a unique solution for minimizing $\mathcal{L}_t^{tr}(\boldsymbol{w}_t^{(m_t)}, \mathcal{D}_t)$, which is determined by the following optimization problem (Evron et al. 2022; S. Lin, Ju, et al. 2023):

$$\min_{\boldsymbol{w}_t} \|\boldsymbol{w}_t - \boldsymbol{w}_{t-1}^{(m_t)}\|_2, \quad \text{s.t. } \mathbf{X}_t^\top \boldsymbol{w}_t = \mathbf{y}_t.$$

- Solving this problem, we update the selected expert m_t for the current task arrival n_t as follows:

$$\boldsymbol{w}_t^{(m_t)} = \boldsymbol{w}_{t-1}^{(m_t)} + \mathbf{X}_t(\mathbf{X}_t^\top \mathbf{X}_t)^{-1}(\mathbf{y}_t - \mathbf{X}_t^\top \boldsymbol{w}_{t-1}^{(m_t)}).$$

- For any other expert $m \in [M]$ not selected (i.e., $m \neq m_t$), its model $\boldsymbol{w}_t^{(m)}$ remains unchanged from $\boldsymbol{w}_{t-1}^{(m)}$.

Training of Gating Network Parameters

After obtaining $\boldsymbol{w}_t^{(m_t)}$, the MoE updates $\boldsymbol{\Theta}_t$ to $\boldsymbol{\Theta}_{t+1}$ using GD.

- On one hand, we aim for $\boldsymbol{\theta}_{t+1}^{(m)}$ of each expert m to specialize in a specific task, which helps mitigate learning loss caused by the incorrect routing of distinct tasks.
- On the other hand, the router needs to balance the load among all experts (Fedus, Zoph, and Shazeer 2022; Li et al. 2024) to reduce the risk of model overfitting and enhance the learning performance in CL.

Key Design I: Multi-objective training loss

- First, we propose the following locality loss function for updating $\mathbf{\Theta}_t$:

$$\mathcal{L}_t^{loc}(\mathbf{\Theta}_t, \mathcal{D}_t) = \sum_{m \in [M]} \pi_m(\mathbf{X}_t, \mathbf{\Theta}_t) \|\boldsymbol{w}_t^{(m)} - \boldsymbol{w}_{t-1}^{(m)}\|_2.$$

- Then we follow the existing MoE literature (e.g., Fedus, Zoph, and Shazeer 2022; Li et al. 2024) to define an auxiliary loss to characterize load balance among the experts:

$$\mathcal{L}_t^{aux}(\mathbf{\Theta}_t, \mathcal{D}_t) = \alpha \cdot M \cdot \sum_{m \in [M]} f_t^{(m)} \cdot P_t^{(m)},$$

where α is constant, $f_t^{(m)} = \frac{1}{t} \sum_{\tau=1}^{t} \mathbb{1}\{m_\tau = m\}$ is the fraction of tasks dispatched to expert m since $t = 1$, and $P_t^{(m)} = \frac{1}{t} \sum_{\tau=1}^{t} \pi_m(\mathbf{X}_\tau, \mathbf{\Theta}_\tau) \cdot \mathbb{1}\{m_\tau = m\}$ is the average probability that the router chooses expert m since $t = 1$.

Key Design I: Multi-objective training loss (cont.)

We finally define the task loss for each task arrival n_t as follows:

$$\mathcal{L}_t^{task}(\mathbf{\Theta}_t, \boldsymbol{w}_t^{(m_t)}, \mathcal{D}_t) = \mathcal{L}_t^{tr}(\boldsymbol{w}_t^{(m_t)}, \mathcal{D}_t) + \mathcal{L}_t^{loc}(\mathbf{\Theta}_t, \mathcal{D}_t) + \mathcal{L}_t^{aux}(\mathbf{\Theta}_t, \mathcal{D}_t).$$

Commencing from the initialization $\mathbf{\Theta}_0$, the gating network is updated based on GD:

$$\boldsymbol{\theta}_{t+1}^{(m)} = \boldsymbol{\theta}_t^{(m)} - \eta \cdot \nabla_{\boldsymbol{\theta}_t^{(m)}} \mathcal{L}_t^{task}(\mathbf{\Theta}_t, \boldsymbol{w}_t^{(m_t)}, \mathcal{D}_t), \forall m \in [M]$$

where $\eta > 0$ is the learning rate.

Key design II: Early termination

Algorithm Training of the MoE model for CL

Input: $T, \sigma_0, \Gamma = \mathcal{O}(\sigma_0^{1.25}), \lambda = \Theta(\sigma_0^{1.25}), I^{(m)} = 0, \alpha = \mathcal{O}(\sigma_0^{0.5}), \eta = \mathcal{O}(\sigma_0^{0.5}), T_1 = \lceil \eta^{-1} M \rceil$;
Initialize $\boldsymbol{\theta}_0^{(m)} = \mathbf{0}$ and $\boldsymbol{w}_0^{(m)} = \mathbf{0}$, $\forall m \in [M]$;
for $t = 1, \cdots, T$ **do**
    Generate $r_t^{(m)}$ for any $m \in [M]$;
    Select m_t and update $\boldsymbol{w}_t^{(m_t)}$;
    if $t > T_1$ **then**
        for $\forall m \in [M]$ with $|h_m - h_{m_t}| < \Gamma$ **do**
            $I^{(m)} = 1$;   // Convergence flag
        end for
    end if
    if $\exists m,$ s.t. $I^{(m)} = 0$ **then**
        Update $\boldsymbol{\theta}_t^{(m)}$ for any $m \in [M]$;
    end if
end for

Theoretical Results: Feature Signal

Lemma 1 ($M > N$ version)

For any two feature matrices $\mathbf{X}$ *and* $\tilde{\mathbf{X}}$ *with the same feature signal* $\boldsymbol{v}_n$*, with probability at least* $1 - o(1)$*, their corresponding gate outputs of the same expert* m *satisfy*

$$\left| h_m(\mathbf{X}, \boldsymbol{\theta}_t^{(m)}) - h_m(\tilde{\mathbf{X}}, \boldsymbol{\theta}_t^{(m)}) \right| = \mathcal{O}(\sigma_0^{1.5}).$$

Given N tasks, all experts can be classified into N sets based on their specialty, where each expert set is defined as:

$$\mathcal{M}_n = \left\{ m \in [M] \middle| n = \arg\max_{j \in [N]} (\boldsymbol{\theta}_t^{(m)})^\top \boldsymbol{v}_j \right\}.$$

Theoretical Results: Convergence of Expert Model

Proposition 1 ($M > N$ version)

Under Algorithm 1, with probability at least $1 - o(1)$*, for any* $t > T_1$*, where* $T_1 = \lceil \eta^{-1} M \rceil$*, each expert* $m \in [M]$ *stabilizes within an expert set* $\mathcal{M}_n$*, and its expert model remains unchanged beyond time* T_1*, satisfying* $\boldsymbol{w}_{T_1+1}^{(m)} = \cdots = \boldsymbol{w}_T^{(m)}$.

Necessity of Early Termination

Proposition 2 ($M > N$ version)

If the MoE keeps updating $\mathbf{\Theta}_t$ at any round $t > T_1$, we obtain:

1. *At round $t_1 = \lceil \eta^{-1}\sigma_0^{-0.25}M \rceil$, the following property holds*

$$\left|h_m(\mathbf{X}_{t_1}, \boldsymbol{\theta}_{t_1}^{(m)}) - h_{m'}(\mathbf{X}_{t_1}, \boldsymbol{\theta}_{t_1}^{(m')})\right| = \begin{cases} \mathcal{O}(\sigma_0^{1.75}), & \text{if } m, m' \in \mathcal{M}_n, \\ \Theta(\sigma_0^{0.75}), & \text{otherwise.} \end{cases}$$

2. *At round $t_2 = \lceil \eta^{-1}\sigma_0^{-0.75}M \rceil$, the following property holds*

$$\left|h_m(\mathbf{X}_{t_2}, \boldsymbol{\theta}_{t_2}^{(m)}) - h_{m'}(\mathbf{X}_{t_2}, \boldsymbol{\theta}_{t_2}^{(m')})\right| = \mathcal{O}(\sigma_0^{1.75}), \forall m, m' \in [M].$$

Benefit of Early Termination

Proposition 3 ($M > N$ version)

Under Algorithm 1, the MoE terminates updating $\mathbf{\Theta}_t$ since round $T_2 = \mathcal{O}(\eta^{-1}\sigma_0^{-0.25}M)$. Then for any task arrival n_t at $t > T_2$, the router selects any expert $m \in \mathcal{M}_{n_t}$ with an identical probability of $\frac{1}{|\mathcal{M}_{n_t}|}$, where $|\mathcal{M}_{n_t}|$ is the number of experts in set $\mathcal{M}_n$.

Definition of Forgetting and Generalization

We define $\mathcal{E}_t(\boldsymbol{w}_t^{(m_t)})$ as the model error in the t-th round:

$$\mathcal{E}_t(\boldsymbol{w}_t^{(m_t)}) = \|\boldsymbol{w}_t^{(m_t)} - \boldsymbol{w}_{n_t}\|_2^2.$$

Following the existing literature on CL (e.g., S. Lin, Ju, et al. 2023; Chaudhry et al. 2018), we assess the performance of MoE in CL using the metrics of forgetting and overall generalization error:

- Forgetting:

$$F_t = \frac{1}{t-1}\sum_{\tau=1}^{t-1}\left(\mathcal{E}_\tau(\boldsymbol{w}_t^{(m_\tau)}) - \mathcal{E}_\tau(\boldsymbol{w}_\tau^{(m_\tau)})\right).$$

- Overall generalization error:

$$G_T = \tfrac{1}{T}\textstyle\sum_{\tau=1}^{T}\mathcal{E}_\tau(\boldsymbol{w}_T^{(m_\tau)}).$$

Benchmark: Performance of Singe Expert

Here we define $r := 1 - \frac{s}{d}$ as the overparameterization ratio.

Proposition 4

If $M = 1$, for any training round $t \in \{2, \cdots, T\}$, we have

$$\mathbb{E}[F_t] = \frac{1}{t-1}\sum_{\tau=1}^{t-1}\left\{\frac{r^t - r^\tau}{N}\sum_{n=1}^{N}\|\boldsymbol{w}_n\|^2 + \frac{r^\tau - r^t}{N^2}\sum_{n\neq n'}\|\boldsymbol{w}_{n'} - \boldsymbol{w}_n\|^2\right\},$$

$$\mathbb{E}[G_T] = \frac{r^T}{N}\sum_{n=1}^{N}\|\boldsymbol{w}_n\|^2 + \frac{1-r^T}{N^2}\sum_{n\neq n'}\|\boldsymbol{w}_n - \boldsymbol{w}'_n\|^2.$$

Performance of MoE

We define $L_t^{(m)} := t \cdot f_t^{(m)}$ as the cumulative number of task arrivals routed to expert m up to round t.

Theorem 1 ($M > N$ Case)

If $M = \Omega(N \ln(N))$, for each round $t \in \{2, \cdots, T_1\}$, the expected forgetting satisfies

$$\mathbb{E}[F_t] < \frac{1}{t-1} \sum_{\tau=1}^{t-1} \Big\{ \frac{r^{L_t^{(m_\tau)}} - r^{L_\tau^{(m_\tau)}}}{N} \sum_{n=1}^{N} \|\boldsymbol{w}_n\|^2 + \frac{r^{L_\tau^{(m_\tau)}} - r^{L_t^{(m_\tau)}}}{N^2} \sum_{n \neq n'} \|\boldsymbol{w}_{n'} - \boldsymbol{w}_n\|^2 \Big\}.$$

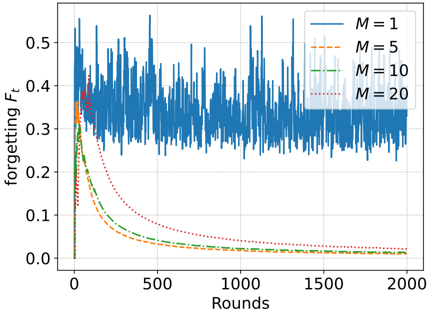
For each $t \in \{T_1 + 1, \cdots, T\}$, we have $\mathbb{E}[F_t] = \frac{T_1 - 1}{t-1} \mathbb{E}[F_{T_1}]$. Further, after training task n_T in the last round T, the overall generalization error satisfies

$$\mathbb{E}[G_T] < \frac{1}{T} \sum_{\tau=1}^{T} \Big\{ \frac{r^{L_{T_1}^{(m_\tau)}}}{N} \sum_{n=1}^{N} \|\boldsymbol{w}_n\|^2 + \frac{1 - r^{L_{T_1}^{(m_\tau)}}}{N^2} \sum_{n \neq n'} \|\boldsymbol{w}_{n'} - \boldsymbol{w}_n\|^2 \Big\}.$$

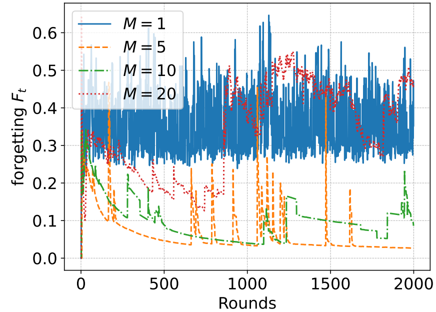
Experiment Setting: Synthetic data

- We first generate N ground truths and their corresponding feature signals.
  - For each $\boldsymbol{w}_n \in \mathbb{R}^d$, we randomly generate d elements by a normal distribution $\mathcal{N}(0, \sigma_0)$. These ground truths are then scaled by a constant to obtain their feature signals $\boldsymbol{v}_n$.
- In each training round t, we generate $(\mathbf{X}_t, \mathbf{y}_t)$ based on ground-truth pool $\mathcal{W}$ and feature signals.
  - After drawing $\boldsymbol{w}_{n_t}$ from $\mathcal{W}$, for $\mathbf{X}_t \in \mathbb{R}^{d\times s}$, we randomly select one out of s samples to fill with $\beta_t \mathbf{v}_{n_t}$. The other $s-1$ samples are generated from $\mathcal{N}(\mathbf{0}, \sigma_t^2 \boldsymbol{I}_d)$.
  - Then, we compute the output $\mathbf{y}_t = \mathbf{X}_t^\top \boldsymbol{w}_{n_t}$.
- Here we set $\sigma_0 = 0.4, \sigma_t = 0.1, d = 10, s = 6, \eta = 0.5, \alpha = 0.5$ and $\lambda = 0.3$.

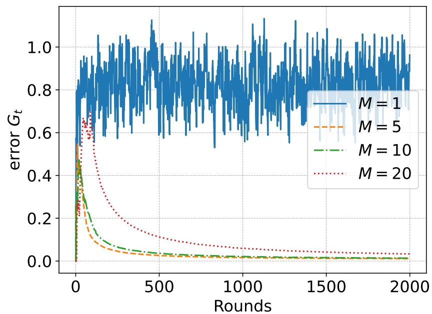
EXPERIMENTS: SYNTHETIC DATA



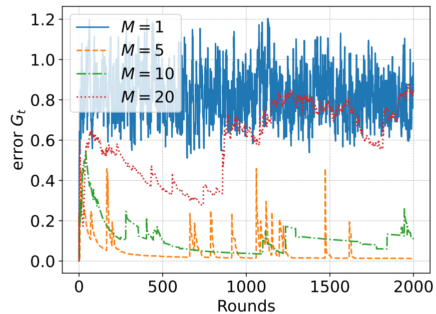
(a) With termination.



(b) Without termination.



(c) With termination.

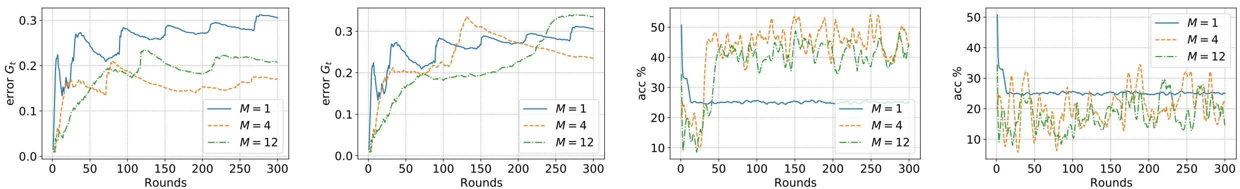


(d) Without termination.

Figure. The dynamics of forgetting and overall generalization errors with and without termination of updating $\mathbf{\Theta}_t$ in Algorithm 1. Here we set $N = 6$ with $K = 3$ clusters and vary $M \in \{1, 5, 10, 20\}$.

Experiments: Real-data Validation

- In each round, we obtain the feature matrix by averaging $s = 100$ training data samples.
- To diversify the model gaps of different tasks, we transform the $d \times d$ matrix into a $d \times d$ dimensional normalized vector to serve as input for the gating network.
- Then we calculate the variance σ_0 of each element across all tasks from the input vector.



(a) With termination. (b) Without termination. (c) With termination. (d) Without termination.

Figure. The dynamics of overall generalization error and test accuracy under the CIFAR-10 dataset (Krizhevsky, Hinton, et al. 2009). Here we set $K = 4, N = 300$ and $M \in \{1, 4, 12\}$.

Conclusion

- We conducted the first theoretical analysis of MoE and its impact on learning performance in CL, focusing on an overparameterized linear regression problem.
- We proved that the MoE model can diversify experts to specialize in different tasks, while its router can learn to select the right expert for each task and balance the loads across all experts.
- Then we demonstrated that, under CL, terminating the updating of gating network parameters after sufficient training rounds is necessary for system convergence.
- Furthermore, we provided explicit forms of the expected forgetting and overall generalization error to assess the impact of MoE.
- Finally, we conducted experiments on real datasets using DNNs to show that certain insights can extend beyond linear models.

References I

Chaudhry, Arslan et al. (2018). **"Efficient lifelong learning with a-gem".** In: *arXiv preprint arXiv:1812.00420*.

Chen, Zixiang et al. (2022). **"Towards Understanding the Mixture-of-Experts Layer in Deep Learning".** In: *Advances in Neural Information Processing Systems* 35, pp. 23049–23062.

Doan, Thang, Seyed Iman Mirzadeh, and Mehrdad Farajtabar (2023). **"Continual learning beyond a single model".** In: *Conference on Lifelong Learning Agents*. PMLR, pp. 961–991.

Du, Nan et al. (2022). **"Glam: Efficient scaling of language models with mixture-of-experts".** In: *International Conference on Machine Learning*. PMLR, pp. 5547–5569.

Evron, Itay et al. (2022). **"How catastrophic can catastrophic forgetting be in linear regression?"** In: *Conference on Learning Theory*. PMLR, pp. 4028–4079.

Fedus, William, Barret Zoph, and Noam Shazeer (2022). **"Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity".** In: *The Journal of Machine Learning Research* 23.1, pp. 5232–5270.

Gao, Rui and Weiwei Liu (2023). **"Ddgr: Continual learning with deep diffusion-based generative replay".** In: *International Conference on Machine Learning*. PMLR, pp. 10744–10763.

Gou, Jianping et al. (2021). **"Knowledge distillation: A survey".** In: *International Journal of Computer Vision* 129.6, pp. 1789–1819.

Hihn, Heinke and Daniel A Braun (2021). **"Mixture-of-Variational-Experts for Continual Learning".** In: *arXiv preprint arXiv:2110.12667*.

References II

- Jin, Xisen et al. (2021). **"Gradient-based editing of memory examples for online task-free continual learning".** In: *Advances in Neural Information Processing Systems* 34, pp. 29193–29205.
- Kirkpatrick, James et al. (2017). **"Overcoming catastrophic forgetting in neural networks".** In: *Proceedings of the National Academy of Sciences* 114.13, pp. 3521–3526.
- Konishi, Tatsuya et al. (2023). **"Parameter-level soft-masking for continual learning".** In: *International Conference on Machine Learning*. PMLR, pp. 17492–17505.
- Krizhevsky, Alex, Geoffrey Hinton, et al. (2009). **"Learning multiple layers of features from tiny images".** In.
- Li, Jing et al. (2024). **"LocMoE: A Low-overhead MoE for Large Language Model Training".** In: *arXiv preprint arXiv:2401.13920*.
- Lin, Bin et al. (2024). **"Moe-llava: Mixture of experts for large vision-language models".** In: *arXiv preprint arXiv:2401.15947*.
- Lin, Sen, Peizhong Ju, et al. (2023). **"Theory on forgetting and generalization of continual learning".** In: *International Conference on Machine Learning*. PMLR, pp. 21078–21100.
- Lin, Sen, Li Yang, et al. (2021). **"TRGP: Trust Region Gradient Projection for Continual Learning".** In: *International Conference on Learning Representations*.
- Rypeść, Grzegorz et al. (2023). **"Divide and not forget: Ensemble of selectively trained experts in Continual Learning".** In: *The Twelfth International Conference on Learning Representations*.

References III

Wang, Liyuan et al. (2022). **"Coscl: Cooperation of small continual learners is stronger than a big one".** In: *European Conference on Computer Vision*. Springer, pp. 254–271.

Yu, Jiazuo et al. (2024). **"Boosting Continual Learning of Vision-Language Models via Mixture-of-Experts Adapters".** In: *arXiv preprint arXiv:2403.11549*.