A. Definitions of Data Heterogeneity in FSSL

We define data heterogeneity in FSSL as follows:

Definition 1 (External heterogeneity in FSSL) *External heterogeneity refers to the distribution discrepancy between $\mathcal{D}_k$ across different clients $\{\mathcal{C}_1, \ldots, \mathcal{C}_K\}$,* i.e.*, for any two different clients $\mathcal{C}_{k_1}$ and $\mathcal{C}_{k_2}$, $\mathcal{P}_{k_1}(\mathbf{Y}) \neq \mathcal{P}_{k_2}(\mathbf{Y})$.*

Definition 2 (Internal heterogeneity in FSSL) *Internal heterogeneity exists within local clients, embodied in: 1) class imbalance, arising from unequal sample sizes among different categories within $\mathcal{C}_k$,* i.e.*, for any two categories c_1 and c_2, $\mathcal{P}_k(\mathbf{Y}(c_1)) \neq \mathcal{P}_k(\mathbf{Y}(c_2))$; 2) distribution imbalance between labeled and unlabeled data, i.e., $\mathcal{P}_k^s(\mathbf{Y}) \neq \mathcal{P}_k^u(\mathbf{Y})$.*

B. Pseudo-Code

The pseudo-code of ProxyFL is shown in Algorithm. 1.

Algorithm 1: Proxy-Guided FSSL (ProxyFL)

Input: Federation system $\mathbb{C}$, communication round T, local learning rate η_l, global model $\mathbf{\Theta}_\mathcal{G}$;

for $t = 0 \rightarrow T-1$ **do**
- Randomly sample a subset of clients $\mathbb{C}_M \subseteq \mathbb{C}$;
- **foreach** *client* $\mathcal{C}_m \in \mathbb{C}_M$ ***in parallel*** **do**
  - Initialize $\mathbf{\Theta}_m^t$ via $\mathbf{\Theta}_\mathcal{G}^t$ from server;
  - Calculate $\mathcal{L}_s$ and $\mathcal{L}_u$ on $\mathbf{x}$ and $\mathbf{u}^{hc}$ via Eq. 9;
  - Calculate $\mathcal{L}_{ICPL}$ on $\mathbf{u}^{hc}$ and $\mathbf{u}^{1c}$ via Eq. 8;
  - $\mathcal{L}_{local} \leftarrow \mathcal{L}_s + \mathcal{L}_u + \mathcal{L}_{ICPL}$;
  - $\mathbf{\Theta}_m^{t+1} \leftarrow \mathbf{\Theta}_m^t - \eta_l \nabla_{\mathbf{\Theta}_m^t} \mathcal{L}_{local}$;
- **end**
- $\theta_\mathcal{G}^{t+1} \leftarrow \sum_{\mathcal{C}_m \in \mathbb{C}_M} \gamma_m \theta_m^{t+1}$, $\mathbf{\Omega}_\mathcal{G} \leftarrow \sum_{\mathcal{C}_m \in \mathbb{C}_M} \gamma_m \omega_m^{t+1}$;
- Server optimizes the global proxy $\mathbf{\Omega}_\mathcal{G}$ via Eq. 2, $\mathbf{\Omega}_\mathcal{G}^{t+1} \leftarrow \mathbf{\Omega}_\mathcal{G}$, thus $\mathbf{\Theta}_\mathcal{G}^{t+1} = \theta_\mathcal{G}^{t+1} \cup \mathbf{\Omega}_\mathcal{G}^{t+1}$;

end

return $\mathbf{\Theta}_\mathcal{G}^T = \theta_\mathcal{G}^T \cup \mathbf{\Omega}_\mathcal{G}^T$

C. Theoretical Proofs

In this section, we provide the convergence analysis for the bi-level optimizations of ProxyFL: Global Proxy Tuning (GPT) and Indecisive-Categories Proxy Learning (ICPL). Our proofs are based on the standard assumptions in the non-convex optimization.

C.1. Convergence of Global Proxy Tuning

Our GPT module is a global optimization process executed on the server. In each communication round, the server collects the local proxies $\{\omega_m\}_{m=1}^M$ from clients and then optimizes the global proxies $\mathbf{\Omega}_\mathcal{G}$ by minimizing the loss function $\mathcal{L}_{GPT}$. First, we give:

Theorem 1 (Convergence of Global Proxy Tuning) *Assume that the loss function $\mathcal{L}_{GPT}$ is L-Lipschitz and bounded below, where $\mathcal{L}_{GPT}$ is related to $\mathbf{\Omega}_\mathcal{G}$. By optimizing the global proxies $\mathbf{\Omega}_\mathcal{G}$ via gradient descent with learning rate $\eta_\mathcal{G}$ such that $0 < \eta_\mathcal{G} \leq \frac{1}{L_\mathcal{G}}$, the optimization process converges to a stationary point. I.e.,*

$$\lim_{Q \to \infty} \frac{1}{Q} \sum_{q=0}^{Q-1} \mathbb{E}\left[\left\| \nabla_{\mathbf{\Omega}_\mathcal{G}} \mathcal{L}_{GPT}\left(\mathbf{\Omega}_\mathcal{G}^q\right) \right\|^2 \right] = 0, \tag{11}$$

where Q is the number of proxy tuning steps on the server.

Then we provide a specific proof for Theorem 1. According to the descent lemma for L-smooth functions, we have:

$$\begin{aligned} \mathcal{L}_{GPT}(\mathbf{\Omega}_\mathcal{G}^{q+1}) \leq & \mathcal{L}_{GPT}(\mathbf{\Omega}_\mathcal{G}^q) + \langle \nabla_{\mathbf{\Omega}_\mathcal{G}} \mathcal{L}_{GPT}(\mathbf{\Omega}_\mathcal{G}^q), \mathbf{\Omega}_\mathcal{G}^{q+1} - \mathbf{\Omega}_\mathcal{G}^q \rangle \\ & + \frac{L_\mathcal{G}}{2} ||\mathbf{\Omega}_\mathcal{G}^{q+1} - \mathbf{\Omega}_\mathcal{G}^q||^2. \end{aligned} \tag{12}$$

According to the gradient-descent formula $\mathbf{\Omega}_\mathcal{G}^{q+1} = \mathbf{\Omega}_\mathcal{G}^q - \eta_\mathcal{G} \nabla_{\mathbf{\Omega}_\mathcal{G}} \mathcal{L}_{GPT}(\mathbf{\Omega}_\mathcal{G}^q)$, Eq. 12 can be re-written as:

$$\begin{aligned} \mathcal{L}_{GPT}(\mathbf{\Omega}_\mathcal{G}^{q+1}) \leq & \mathcal{L}_{GPT}(\mathbf{\Omega}_\mathcal{G}^q) - \eta_\mathcal{G} ||\nabla_{\mathbf{\Omega}_\mathcal{G}} \mathcal{L}_{GPT}(\mathbf{\Omega}_\mathcal{G}^q)||^2 \\ & + \frac{L_\mathcal{G} \eta_\mathcal{G}^2}{2} ||\nabla_{\mathbf{\Omega}_\mathcal{G}} \mathcal{L}_{GPT}(\mathbf{\Omega}_\mathcal{G}^q)||^2 \end{aligned} \tag{13}$$

Then, the right side will be:

$$\mathcal{L}_{GPT}(\mathbf{\Omega}_\mathcal{G}^{q+1}) \leq \mathcal{L}_{GPT}(\mathbf{\Omega}_\mathcal{G}^q) - \eta_\mathcal{G}(1 - \frac{L_\mathcal{G} \eta_\mathcal{G}}{2}) ||\nabla_{\mathbf{\Omega}_\mathcal{G}} \mathcal{L}_{GPT}(\mathbf{\Omega}_\mathcal{G}^q)||^2 \tag{14}$$

Let the learning rate $\eta_\mathcal{G} \leq \frac{1}{L_\mathcal{G}}$, such that $1 - \frac{L_\mathcal{G} \eta_\mathcal{G}}{2} \geq \frac{1}{2}$. Thus, Eq. 13 can be simplified to:

$$\mathcal{L}_{GPT}(\mathbf{\Omega}_\mathcal{G}^{q+1}) \leq \mathcal{L}_{GPT}(\mathbf{\Omega}_\mathcal{G}^q) - \frac{\eta_\mathcal{G}}{2} ||\nabla_{\mathbf{\Omega}_\mathcal{G}} \mathcal{L}_{GPT}(\mathbf{\Omega}_\mathcal{G}^q)||^2 \tag{15}$$

Rearranging the terms, we get:

$$||\nabla_{\mathbf{\Omega}_\mathcal{G}} \mathcal{L}_{GPT}(\mathbf{\Omega}_\mathcal{G}^q)||^2 \leq \frac{2}{\eta_\mathcal{G}} (\mathcal{L}_{GPT}(\mathbf{\Omega}_\mathcal{G}^q) - \mathcal{L}_{GPT}(\mathbf{\Omega}_\mathcal{G}^{q+1})) \tag{16}$$

Summing both the left and right sides of Eq. 16 from $q = 0$ to $Q-1$, we have:

$$\begin{aligned} \sum_{q=0}^{Q-1} \|\nabla_{\mathbf{\Omega}_\mathcal{G}} \mathcal{L}_{GPT}(\mathbf{\Omega}_\mathcal{G}^q)\|^2 \leq & \underbrace{\frac{2}{\eta_\mathcal{G}} \sum_{q=0}^{Q-1} (\mathcal{L}_{GPT}(\mathbf{\Omega}_\mathcal{G}^q) - \mathcal{L}_{GPT}(\mathbf{\Omega}_\mathcal{G}^{q+1}))}_{} \\ = & \frac{2}{\eta_\mathcal{G}} (\mathcal{L}_{GPT}(\mathbf{\Omega}_\mathcal{G}^0) - \mathcal{L}_{GPT}(\mathbf{\Omega}_\mathcal{G}^Q)) \end{aligned} \tag{17}$$

Since we adopt InfoNCE loss [27] for $\mathcal{L}_{GPT}$ with its lower-bound 0, we thus have:

$$\sum_{q=0}^{Q-1} \|\nabla_{\mathbf{\Omega}_\mathcal{G}} \mathcal{L}_{GPT}(\mathbf{\Omega}_\mathcal{G}^q)\|^2 \leq \frac{2}{\eta_\mathcal{G}} \mathcal{L}_{GPT}(\mathbf{\Omega}_\mathcal{G}^0) \tag{18}$$

Table 6. Experimental results on CIFAR-10, CIFAR-100, SVHN and CINIC-10 under 20% label. Bold text indicates the best result, and the last row presents the improvement of ProxyFL over the second best method.

Methods	CIFAR-10			CIFAR-100			SVHN			CINIC-10		
	$\alpha=0.1$	$\alpha=0.5$	$\alpha=1$	$\alpha=0.1$	$\alpha=0.5$	$\alpha=1$	$\alpha=0.1$	$\alpha=0.5$	$\alpha=1$	$\alpha=0.1$	$\alpha=0.5$	$\alpha=1$
FL Methods												
FedAvg [23]	86.37	87.06	87.97	45.72	46.57	47.55	88.37	89.05	89.97	66.24	68.29	69.21
FedProx [17]	86.78	88.11	88.44	45.96	47.33	47.89	87.99	88.56	91.10	65.53	69.57	69.91
FedAvg-SL	90.46	91.24	91.32	67.98	68.83	69.10	94.11	94.41	94.40	77.82	80.42	81.29
FL+SSL Methods												
FixMatch-LPL	87.22	89.61	89.23	56.80	57.35	57.59	93.66	94.11	94.21	72.51	75.14	76.03
FixMatch-GPL	88.55	89.69	89.83	57.02	57.85	57.85	93.89	94.12	94.17	76.14	77.35	77.82
FedProx+FixMatch	87.47	89.46	89.56	57.44	57.91	57.87	93.60	93.93	94.05	72.36	75.15	76.06
FedAvg+FlexMatch	76.36	78.66	78.76	58.24	58.44	58.79	56.94	58.58	62.19	73.32	75.75	75.95
FSSL Methods												
FedMatch [12]	82.44	84.13	85.21	45.07	47.29	48.40	93.01	93.58	93.76	66.94	68.60	72.34
FedLabel [4]	87.37	88.86	88.93	58.63	58.98	59.23	93.44	94.38	94.59	60.13	67.30	72.22
FedLoke [36]	84.57	85.26	86.98	53.87	53.67	54.56	93.26	93.45	93.57	70.63	71.61	71.78
FedDure [1]	88.56	89.63	89.95	56.14	57.23	57.89	93.81	94.42	94.37	76.21	77.13	77.75
FedDB [39]	85.19	86.36	86.65	52.81	54.62	55.48	93.22	93.50	94.27	74.18	75.00	75.65
SAGE [22]	89.87	90.53	90.54	60.86	61.49	62.01	94.31	94.56	94.68	77.51	78.23	78.77
ProxyFL (ours)	**90.97**	**91.22**	**91.51**	**62.57**	**63.09**	**63.19**	**95.03**	**95.40**	**95.34**	**80.80**	**81.02**	**81.46**
	↑ 1.10	↑ 0.69	↑ 0.97	↑ 1.71	↑ 1.60	↑ 1.18	↑ 0.72	↑ 0.84	↑ 0.66	↑ 3.29	↑ 2.79	↑ 2.69

Dividing both sides by Q and taking the limit, we have:

$$\lim_{Q\to\infty}\frac{1}{Q}\sum_{q=0}^{Q-1}||\nabla_{\mathbf{\Omega}_\mathcal{G}}\mathcal{L}_{GPT}(\mathbf{\Omega}_\mathcal{G}^q)||^2 \leq \lim_{Q\to\infty}\frac{2\mathcal{L}_{GPT}(\mathbf{\Omega}_\mathcal{G}^0)}{\eta_\mathcal{G} Q}=0 \tag{19}$$

Since the squared norm of the gradient (*i.e., the left-hand side of Eq. 19*) is non-negative, **hence we have proven Theorem 1.**

C.2. Convergence of Local Training with ICPL

The ICPL module is executed during the local training on each client. The total local loss is denoted as $\mathcal{L}_{local} = \mathcal{L}_s + \alpha\mathcal{L}_u + \beta\mathcal{L}_{ICPL}$, with optimization parameters $\mathbf{\Theta}_k = \theta_k \cup \omega_k$.

Theorem 2 (Convergence of Local Training with ICPL) *Suppose that the total local loss function $\mathcal{L}_k$ for client k is L-smooth and bounded below, where $\mathcal{L}_k$ is related to $\mathbf{\Theta}_k$. By optimizing the local model parameters $\mathbf{\Theta}_k$ via gradient descent with learning rate η_l such that $0<\eta_l\leq\frac{1}{L_k}$, the optimization process converges to a stationary point. I.e.,*

$$\lim_{E\to\infty}\frac{1}{E}\sum_{e=0}^{E-1}\mathbb{E}[||\nabla_{\mathbf{\Theta}_k}\mathcal{L}_k(\mathbf{\Theta}_k^e)||^2]=0 \tag{20}$$

where E is the number of local training epochs.

Similar to Sec. C.1, following the descent lemma and the gradient-descent update formula, we can similarly derive:

$$\mathcal{L}_k(\mathbf{\Theta}_k^{e+1}) \leq \mathcal{L}_k(\mathbf{\Theta}_k^e) - \eta_l(1-\frac{L_k\eta_l}{2})\|\nabla_{\mathbf{\Theta}_k}\mathcal{L}_k(\mathbf{\Theta}_k^e)\|^2 \tag{21}$$

By setting a local learning rate $\eta_l \leq \frac{1}{L_k}$, we obtain:

$$\begin{aligned}\sum_{e=0}^{E-1}||\nabla_{\Theta_k}\mathcal{L}_k(\mathbf{\Theta}_k^e)||^2 &\leq \frac{2}{\eta_l}(\mathcal{L}_k(\mathbf{\Theta}_k^0)-\mathcal{L}_k(\mathbf{\Theta}_k^E))\\ &\leq \frac{2}{\eta_l}\mathcal{L}_k(\mathbf{\Theta}_k^0)\end{aligned} \tag{22}$$

Taking the limit to Eq. 22 and knowing that the squared norm is non-negative, we have:

$$0\leq\lim_{E\to\infty}\frac{1}{E}\sum_{e=0}^{E-1}||\nabla_{\mathbf{\Theta}_k}\mathcal{L}_k(\mathbf{\Theta}_k^e)||^2 \leq \lim_{E\to\infty}\frac{2}{\eta_l E}\mathcal{L}_k(\mathbf{\Theta}_k^0)=0 \tag{23}$$

Thus,

$$\lim_{E\to\infty}\frac{1}{E}\sum_{e=0}^{E-1}||\nabla_{\mathbf{\Theta}_k}\mathcal{L}_k(\mathbf{\Theta}_k^e)||^2=0 \tag{24}$$

So we have proven Theorem 2.

D. Additional Empirical Analysis

We provide some additional experiments to complement Sec. 6 of our main paper, thereby offering a more robust demonstration of our proposed ProxyFL.

D.1. Labeling Ratio

To validate the effectiveness of our ProxyFL, we give a comparison between ProxyFL and other state-of-the-art methods with a 10% labeling ratio in Tab. 1 of main paper.

Table 7. The communication costs per round between different methods in our experimental setting as SAGE [22]. $\mathbf{\Theta}$ represents model parameters. σ and ψ denotes the parameters after sparsely decomposition in FedMatch [12], and H means the number of helper agents. C and d means respectively the number of categories and feature dimension.

	Formulation	Practice
FixMatch+FSSL	$\sum_{m=1}^{M} \lvert\mathbf{\Theta}\rvert + \lvert\mathbf{\Theta}\rvert$	22.08 M
FedMatch	$\sum_{i=1}^{M}(\lvert\Delta\sigma_i\rvert + \lvert\Delta\psi_i\rvert + C) + (\lvert\Delta\sigma_{\mathcal{G}}\rvert + \lvert\Delta\psi_{\mathcal{G}}\rvert + H \times \lvert\psi_h\rvert)$	33.12 M
FedLabel	$\sum_{m=1}^{M} \lvert\mathbf{\Theta}\rvert + \lvert\mathbf{\Theta}\rvert$	22.08 M
FedDure	$\sum_{m=1}^{M} \lvert\mathbf{\Theta}\rvert + \lvert\mathbf{\Theta}\rvert$	22.08 M
FedDB	$\sum_{i=1}^{M}(\lvert\mathbf{\Theta}\rvert + C) + \lvert\mathbf{\Theta}\rvert$	22.08 M
SAGE	$\sum_{m=1}^{M} \lvert\mathbf{\Theta}\rvert + \lvert\mathbf{\Theta}\rvert$	22.08 M
Proto. + FSSL	$\sum_{m=1}^{M} \lvert\mathbf{\Theta}\rvert + \lvert\mathbf{\Theta}\rvert + C \times d$	22.90 M
ProxyFL	$\sum_{i=1}^{M}(\lvert\mathbf{\Theta}\rvert + C) + (\lvert\mathbf{\Theta}\rvert + C)$	22.08 M

Table 8. Comparison of convergence rates between ProxyFL and other baseline methods on CIFAR100 dataset with $\alpha = 0.5$ (the upper part) and $\alpha = 1$ (the lower part).

Acc.	30%		40%		50%	
Method	Round↓	Speedup↑	Round↓	Speedup↑	Round↓	Speedup↑
LPL	121	×1.00	221	×1.00	546	×1.00
GPL	113	×1.07	210	×1.05	419	×1.30
FedLabel	83	×1.46	160	×1.38	366	×1.49
FedDB	94	×1.29	205	×1.08	492	×1.11
FedDure	110	×1.10	222	×1.00	552	×0.99
SAGE	55	×2.20	105	×2.10	241	×2.27
ProxyFL	**44**	**×2.75**	**79**	**×2.80**	**155**	**×3.52**
LPL	118	×1.00	267	×1.00	527	×1.00
GPL	94	×1.26	183	×1.46	390	×1.35
FedLabel	91	×1.30	164	×1.63	341	×1.55
FedDB	103	×1.15	237	×1.13	418	×1.26
FedDure	95	×1.24	182	×1.47	450	×1.17
SAGE	56	×2.11	112	×2.38	242	×2.18
ProxyFL	**45**	**×2.62**	**83**	**×3.22**	**157**	**×3.36**

Here, we further study the robustness of ProxyFL by comparing ProxyFL with other methods at 20% labeling ratio. As shown in Tab. 6, ProxyFL consistently achieves better performance across different labeling ratios than others.

D.2. Communication Costs

We calculate the communication cost in both theory and practice. As shown in Tab. 7, FedMatch [12] needs the most communication costs of some auxiliary parameters since they additionally uploads the model embedding for KD-Tree reconstruction and downloads H helper agents to facilitate local training. Incorporating 'prototypes' into FSSL will bring extra costs about 0.82 M as $\sum_{m=1}^{M} C \times d$ and high-dimensional features (prototypes) poses the risk of reversely reconstruction [7, 24]. As shown in Tab. 7, compared to [1, 4, 22], the costs of $\mathcal{P}'_{\mathcal{G}}(\mathbf{Y})$ in ProxyFL could be negligible ($\sum_{m=1}^{M} C + C$, approximately 0.0016 M, similar to [39]) and our method achieves much higher accuracy than them. In summary, our ProxyFL achieves higher accuracy while preserving a reasonable communication efficiencies and our prior design has lower privacy-leakage risk [39] than raw samples or features.

Table 9. Performance gains brought by ProxyFL as a plugin to other baseline methods.

Methods	CIFAR-10	CIFAR-100	SVHN	CINIC-10
FixMatch	82.98	49.32	89.68	68.02
+ProxyFL	88.56	57.50	95.09	77.98
	↑ 5.58	↑ 8.18	↑ 5.41	↑ 9.96
FedLabel	62.85	50.88	89.31	67.64
+ProxyFL	88.55	56.88	94.94	76.35
	↑ 25.70	↑ 6.00	↑ 5.63	↑ 8.71
SAGE	87.05	54.18	93.85	74.59
+ProxyFL	88.29	57.10	95.01	77.58
	↑ 1.24	↑ 2.92	↑ 1.16	↑ 2.99

D.3. Convergence Rate

In the 2) of Sec. 6.3 of the main paper, we conduct experiments under the $\alpha = 0.1$ setting, where the ProxyFL method significantly improve model convergence speed and test accuracy. Here, we provide a detailed comparison of SAGE and other methods under varying levels of heterogeneity. As demonstrated in Tab. 8 and theoretical proofs of Sec. C, ProxyFL could achieve substantial acceleration in early stages and ensure final convergence under different levels of heterogeneity.

D.4. ProxyFL as a Plug-in Approach

The GPT and ICPL components in our ProxyFL can be considered as a optimization mechanism of category distribution for FSSL, allowing integration into other FSSL approaches. As present in Tab. 9, ProxyFL consistently improves the performance of existing FSSL methods. With a proxy-guided mechanism, our approach alleviates both internal and external heterogeneity locally and globally.

D.5. Ablation Study on Tuning Epochs

We conduct experiments for server-tuning epochs, *i.e.*, the hyper-parameter Q. As shown in Fig. 7, our GPT module achieves best performance when $Q \approx 100$ across most datasets. But, for CIFAR100 dataset, it requires fewer tuning epochs (*i.e.*, $Q \approx 10$) and exhibits overfitting when Q increases. We analyze that, since CIFAR100 dataset has more categories ($\times 10$ times category-number than other datasets) and thus has more tuning samples, it will converge much faster than other datasets. Thus, corresponding to Sec. 6.1 of our main paper, we claim that the number of tuning epochs is set to 10 for CIFAR-100 dataset and 100 for the other datasets.

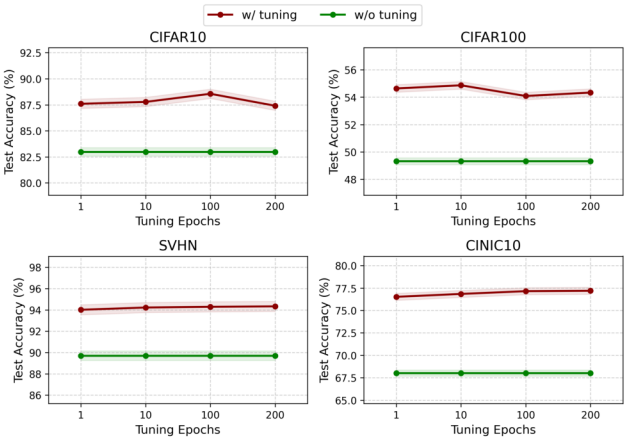

Figure 7. Ablation on the numbers of epoch for Global Proxy Tuning.

D.6. Experimental Setting and other Details

As claimed in Sec. 6.1 of our main paper, **we strictly follow the experimental setup of SAGE [22].** We conduct our experiments on the same datasets with consistent client-splitting strategies as SAGE. For the hyper-parameter 'the communication rounds T', we follow SAGE to set $300, 500, 150, 400$ rounds for CIFAR10, CIFAR100, SVHN and CINIC10 dataset, respectively. The local learning rate is set to 0.1 and the confidence threshold τ for pseudo-labeling set to 0.95, *etc.* In summary, unless otherwise specified, our experimental details are consistent with SAGE.