A APPENDIX

A ALGORITHM PROCEDURE

Algorithm 1 outlines the procedure of margin selection (MS). In MS, distances of the current sample $(\boldsymbol{x}, y)$ to each other class c are computed. If $y \neq c$, the classification margin of $(\boldsymbol{x}, y)$ and class c is $M(\boldsymbol{x}, y, c)$, which is the distance of moving $\boldsymbol{x}$ from class y to class c. If $y = c$, the classification margin is $\min_{\tilde{c} \neq y} M(\boldsymbol{x}, y, \tilde{c})$, which corresponds to the distance moving $(\boldsymbol{x}, y)$ to another class that is the most close to $\boldsymbol{x}$. For the whole candidate set $\mathcal{T}$, this generates a $|\mathcal{T}| \times C$ score matrix. After the classification margins are obtained, $|\mathcal{S}|/C$ samples with the smallest classification margin along each class are picked. This keeps samples collected in the subset balanced.

Algorithm 1 Margin selection: $\mathrm{MS}(\mathrm{w}, \mathcal{T}, \gamma)$

Input:
Candidate set $\mathcal{T}$, keeping ratio γ, number of classes C;
Network with weights $\boldsymbol{w}$, including weights of the final classification layer $\boldsymbol{W}$;
Output:
Selected subset according to the classification margin $\mathcal{S}$.
Compute the keeping budget $|\mathcal{S}| = \gamma \cdot |\mathcal{T}|$, initialize the subset $\mathcal{S} = \{\}$
// Evaluating: compute the classification margin.
for $(\boldsymbol{x}, y) \in \mathcal{T}$ **do**
  for $c = 1 : C$ **do**
    Compute the classification margin of the sample to the $(\boldsymbol{y}, c)$ boundary:

$$M(\boldsymbol{x}, y, c) = \begin{cases} \min_{\tilde{c} \neq y} M(\boldsymbol{x}, y, \tilde{c}) & y = c \\ M(\boldsymbol{x}, y, c) & y \neq c \end{cases} \tag{4}$$

  end for
end for
// Selecting: pick the samples according to classification margin (Equation 4)
for $c = 1 : C$ **do**
  Pick $|\mathcal{S}|/C$ samples which have the smallest classification margins ($M(\cdot)$): $\mathrm{Top}_{|\mathcal{S}|/C}(c)$.
  $\mathcal{S} = \mathcal{S} \bigcup \mathrm{Top}_{|\mathcal{S}|/C}(c)$
  Remove the already selected samples from the candidate set: $\mathcal{T} = \mathcal{T} - \mathrm{Top}_{|\mathcal{S}|/C}(c)$
end for

Algorithm 2 Dynamic margin selection (DynaMS)

Input:
Training data $\mathcal{T}$;
Base network with weights $\boldsymbol{\mathcal{W}}$, learning rate η
Keep ratio of each selection γ_k where $k = 1, ..., K$, selection interval Q
Output:
Model efficiently trained on selected subsets.
$k = 1$; $\gamma_k = 1$ thus $\mathcal{S}_k = \mathcal{T}$
for epochs $t = 1, ..., T$ **do**
  if $t \ \% \ Q == 0$ **then**
    Select subset, $\mathcal{S}_k = \mathrm{MS}(\boldsymbol{\mathcal{W}}^t, \mathcal{T}, \gamma_k)$.
    $k = k + 1$
  else
    Keep subset $\mathcal{S}_k$.
  end if
  Update $\boldsymbol{\mathcal{W}}$ via stochastic gradient descent on $\mathcal{S}_k$.
end for

Algorithm 3 Dynamic margin selection (DynaMS) with parameter sharing proxy (PSP)

Input:
Training data $\mathcal{T}$;
Base network with weights $\boldsymbol{\mathcal{W}}$, learning rate η
Keep ratio of each selection γ_k where $k = 1, ..., K$, selection interval Q
Slimming factor of the proxy r, thus the proxy weights $\boldsymbol{\mathcal{W}}_{\text{proxy}}$ is determined.
Output:
Model efficiently trained on selected subsets.

$k = 1$; $\gamma_k = 1$ thus $\mathcal{S}_k = \mathcal{T}$
for epochs $t = 1, ..., T$ **do**
  if $t \ \% \ Q == 0$ **then**
    Select subset, $\mathcal{S}_k = \text{MS}(\boldsymbol{\mathcal{W}}^t_{\text{proxy}}, \mathcal{T}, \gamma_k)$.
    $k = k + 1$
  else
    Keep subset $\mathcal{S}_k$.
  end if
  Update $\boldsymbol{\mathcal{W}}$ via optimizing $\mathcal{L}(\boldsymbol{\mathcal{W}}) + \mathcal{L}(\boldsymbol{\mathcal{W}}_{\text{proxy}})$ on $\mathcal{S}_k$. (Slimmable training)
end for

A full workflow of efficient training with the proposed dynamic margin selection (DynaMS) is shown in Algorithm 2. The model is first trained on the full dataset $\mathcal{T}$ for Q epochs to warm up. Subset selection kicks in each Q epochs, samples are evaluated with the current model so the informative subset gets updated according to the distance of samples to the classification boundary. After selection, the model is trained on the selected subset until the next selection. The workflow incorporating parameter sharing proxy is shown in Algorithm 3. Different from naive DynaMS, samples are evaluated and selected with the proxy instead of the underlying model. During the Q epochs' training, the proxy and the original model are updated simultaneously with slimmable training (Yu et al., 2019).

B Proof for Theorem 2.2

To prove Theorem 2.2, we first inspect the norm of $\boldsymbol{x}$. We get the following lemma.

Lemma 1. *For Gaussian data $\boldsymbol{x} \sim \mathcal{N}(0, \Sigma)$, let $\mu > 0$, $T > 1$ be constants, d the dimension of $\boldsymbol{x}$ and λ the largest eigenvalue of the covariance Σ, then with probability at least $1 - \frac{1}{\mu T d}$, $\|\boldsymbol{x}\|_2 < \sqrt{d\lambda}(1 + (2\mu)^{\frac{1}{4}})T^{\frac{1}{4}}$.*

Proof of Lemma 1. For $\boldsymbol{x} \sim \mathcal{N}(0, \Sigma)$, $\|\boldsymbol{x}\|_2^2$ follows a generalized chi-squared distribution. The mean and variance can be computed explicitly as $\mathbb{E}[\|\boldsymbol{x}\|_2^2] = \text{tr}\Sigma = \sum_j \lambda_j$ and $\text{Var}(\|\boldsymbol{x}\|_2^2) = 2\text{tr}\Sigma^2 = 2\sum_j \lambda_j^2$. By Chebyshev's inequality, we have

$$Pr\left(\|\boldsymbol{x}\|_2^2 < \sum \lambda_j + \sqrt{\mu T d}\sqrt{2\sum_j \lambda_j^2}\right) > 1 - \frac{1}{\mu T d}$$

where $\mu > 0$ and $T > 1$ are constants and d is the dimension of x. Then

as $\sum \lambda_j + \sqrt{\mu T d}\sqrt{2\sum_j \lambda_j^2} \leq (1 + \sqrt{2\mu T})d\lambda$ where $\lambda = \max_j \lambda_j$ is the largest eigenvalue of the covariance Σ, we have:

$$Pr\left(\|\boldsymbol{x}\|_2 < \sqrt{d\lambda}(1 + (2\mu)^{\frac{1}{4}})T^{\frac{1}{4}}\right) > 1 - \frac{1}{\mu T d} \tag{5}$$

□

Then we can start proving Theorem 2.2.

Theorem. *Consider logistic regression $f(\boldsymbol{x}) = \frac{1}{1+e^{-\boldsymbol{w}^\top \boldsymbol{x}}}$ with N Gaussian training samples $\boldsymbol{x} \sim \mathcal{N}(0, \Sigma)$, $\boldsymbol{x} \in \mathbb{R}^d$. Assume $\|\boldsymbol{w}\|_2 \leq D$ and $\frac{N}{d} < \alpha$. Let $\boldsymbol{w}^*$ be the optimal parameters and λ be*

the largest eigenvalue of the covariance Σ. *For* $t \in \{1, \dots T\}$ *and constants* $\varepsilon > D\sqrt{\frac{\lambda}{2}} - 1, \zeta > 1, \mu >> \alpha$, *select subset with critical margin* $\kappa_t = (1+\varepsilon)\log(\zeta T - t)$ *and update parameters with learning rate* $\eta = \frac{DN}{E\sqrt{T}}$. *Then with probability at least* $1 - \frac{\alpha}{\mu}$

$$\min_t \mathcal{L}(\boldsymbol{w}_t) - \mathcal{L}(\boldsymbol{w}^*) \leq DE\left(\frac{1}{T^{\frac{1}{4}}} + \frac{c_{\varepsilon,\zeta}}{T^{\frac{3}{4}+\varepsilon}} + \frac{c_{\varepsilon,\zeta,\lambda}}{T^{\beta}}\right) \tag{6}$$

where $E = \sqrt{d\lambda}(1 + (2\mu)^{\frac{1}{4}})$, $\beta = \frac{(1+\varepsilon)^2}{2D^2\lambda} - \frac{1}{4}$, $c_{\varepsilon,\zeta}$ *and* $c_{\varepsilon,\zeta,\lambda}$ *are constants depending on* ε, ζ *and* λ.

Proof of Theorem 2.2. For logistic regression $f(\boldsymbol{x}) = \frac{1}{1+e^{-\boldsymbol{w}^\top \boldsymbol{x}}}$ with loss function

$$\mathcal{L} = \frac{1}{N}\sum_{i=1}^{N} \ell_i = \frac{1}{N}\sum_{i=1}^{N} -y_i \log \hat{y}_i - (1-y_i)\log(1-\hat{y}_i) \tag{7}$$

Where $\hat{y}_i$ is the predicted value. The gradient incurred training on the selected subset is then:

$$\frac{\partial \mathcal{L}_\kappa}{\partial \boldsymbol{w}} = \frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i - y_i)\boldsymbol{x}_i \cdot \mathbf{I}(|\boldsymbol{w}^\top \boldsymbol{x}_i| < \kappa)$$

For those $|\boldsymbol{w}^\top x_i| \geq \kappa$ or "easy" samples, we have $|\operatorname{sgn}(y_i - \frac{1}{2}) \cdot \boldsymbol{w}^\top \boldsymbol{x}_i| \geq \kappa$ and with probability at least $1 - \frac{1}{\mu T d}$

$$\left\| \frac{\partial \ell_i}{\partial \boldsymbol{w}} \right\|_2 \leq \begin{cases} \frac{E \cdot T^{\frac{1}{4}}}{1+e^{\kappa}} & \text{if } \operatorname{sgn}(y_i - \frac{1}{2}) \cdot \boldsymbol{w}^\top \boldsymbol{x}_i \geq \kappa \\ E \cdot T^{\frac{1}{4}} & \text{if } \operatorname{sgn}(y_i - \frac{1}{2}) \cdot \boldsymbol{w}^\top \boldsymbol{x}_i \leq -\kappa \end{cases} \tag{8}$$

where $E = \sqrt{d\lambda}(1 + (2\mu)^{\frac{1}{4}})$. Note that the condition $\operatorname{sgn}(y_i - \frac{1}{2}) \cdot \boldsymbol{w}^\top x_i \leq -\kappa$ means x_i is misclassified by $\boldsymbol{w}$ as well as the margin is at least κ. Denote the portion of this kind of misclassified sample in the whole training set by r, we have the estimate of the gradient gap

$$\begin{aligned} \operatorname{Err}_t = &\left\| \frac{\partial \mathcal{L}_\kappa}{\partial \boldsymbol{w}} - \frac{\partial \mathcal{L}}{\partial \boldsymbol{w}} \right\|_2 = \frac{1}{N}\left\| \sum_{|\boldsymbol{w}^\top \boldsymbol{x}| \geq \kappa} \frac{\partial \ell}{\partial \boldsymbol{w}}(\boldsymbol{x}) \right\|_2 \\ \leq &\frac{ET^{\frac{1}{4}}(1-\gamma_t)}{1+e^{\kappa_t}} + ET^{\frac{1}{4}}(1-\gamma_t)r_t \end{aligned} \tag{9}$$

Where γ_t is the fraction of data kept by selecting with margin κ_t. The inequality holds with probability at least $(1 - \frac{1}{\mu T d})^N > 1 - \frac{\alpha}{\mu T}$ because of Equation 8.

Note that Lemma 1 also suggest $\left\| \frac{\partial \ell}{\partial \boldsymbol{w}} \right\|_2 \leq E \cdot T^{\frac{1}{4}}$ with large probability, therefore $\mathcal{L}$ is highly likely to be Lipschitz continuous with parameter $ET^{\frac{1}{4}}$. By setting a constant learning rate $\eta = \frac{DN}{E\sqrt{T}}$, and critical margin $\kappa_t = (1+\varepsilon)\log(\zeta T - t), \zeta > 1$, we have with probability at least $\left(1 - \frac{\alpha}{\mu T}\right)^T \geq 1 - \frac{\alpha}{\mu}$

$$\begin{aligned} \min_t \mathcal{L}(\boldsymbol{w}_t) - \mathcal{L}(\boldsymbol{w}^*) &\leq \frac{DE}{NT^{\frac{1}{4}}} + \frac{D}{T}\sum_{t=1}^{T-1} \operatorname{Err}_t \\ &\leq \frac{DE}{NT^{\frac{1}{4}}} + \frac{DE}{T^{\frac{3}{4}}}\sum_{t=1}^{T-1} \frac{1}{(\zeta T - t)^{1+\varepsilon}} + \frac{DE}{T^{\frac{3}{4}}}\sum_{t=1}^{T-1} r_t \\ &\leq \frac{DE}{T^{\frac{1}{4}}}\left(\frac{1}{N} + \frac{c_{\varepsilon,\zeta}}{T^{\varepsilon}\sqrt{T}}\right) + \frac{DE}{T^{\frac{3}{4}}}\sum_{t=1}^{T-1} r_t \end{aligned} \tag{10}$$

The first inequality follows the Theorem 1 in (Killamsetty et al., 2021). The last inequality holds because $\sum_{t=1}^{T-1} \frac{1}{(\zeta T - t)^{1+\varepsilon}} \leq \int_{(\zeta-1)T}^{\zeta T} \frac{1}{s^{1+\varepsilon}} ds \leq \frac{c_{\varepsilon,\zeta}}{T^{\varepsilon}}$ with $c_{\varepsilon,\zeta} = \frac{1}{\varepsilon(\zeta-1)^{\varepsilon}}, \forall \varepsilon > 0$ and $\zeta > 1$.

To bound the sum of classification error (the last term of Equation 10), again we utilize the data distribution prior. Note that the data points contribute to r are quantified by the following set:

$$E = \{\boldsymbol{w}_o^\top \boldsymbol{x} > 0 \wedge \boldsymbol{w}^\top \boldsymbol{x} < -\kappa\} \cup \{\boldsymbol{w}_o^\top \boldsymbol{x} < 0 \wedge \boldsymbol{w}^\top \boldsymbol{x} > \kappa\} := E_1 \cup E_2$$

where $\boldsymbol{w}_o$ is the oracle classifier such that the true label is generated according to $y = \text{sgn}(\boldsymbol{w}_o^\top \boldsymbol{x})$. Let ϕ represent the probability density function of standard Gaussian, we see that

$$\begin{aligned} r =& \int_E \phi(x|\Sigma)dx = 2\int_{E_1} \phi(x|\Sigma)dx \\ \leq& 2\int_{\{\boldsymbol{w}^\top \cdot x < -\kappa\}} \phi(x|\Sigma)dx = 2\Phi\bigg(-\frac{\kappa}{\sqrt{\boldsymbol{w}^\top \Sigma \boldsymbol{w}}}\bigg) \\ \leq& 2\Phi\bigg(-\frac{\kappa}{D\sqrt{\lambda}}\bigg) \end{aligned}$$

where λ is the largest eigenvalue of Σ. Therefore, we have the following estimation:

$$\begin{aligned} \frac{1}{T^{\frac{3}{4}}}\sum_{t=1}^{T-1} r_t \leq& \frac{1}{T^{\frac{3}{4}}}\sum_{t=1}^{T-1} 2\Phi\bigg(-\frac{\kappa_t}{D\sqrt{\lambda}}\bigg) \\ \leq& \frac{2}{T^{\frac{3}{4}}}\sum_{t=1}^{T-1}\frac{\phi(\kappa_t/(D\sqrt{\lambda}))}{\kappa_t/(D\sqrt{\lambda})} \qquad \text{(Gaussian upper tail bound)} \\ =& \frac{2D\sqrt{\lambda}}{\sqrt{2\pi}(1+\varepsilon)}\frac{1}{T^{\frac{3}{4}}}\sum_{t=1}^{T-1}\frac{1}{\log(\zeta T - t)} e^{-\frac{(1+\varepsilon)^2}{2D^2\lambda}\log^2(\zeta T - t)} \\ \leq& \frac{2D\sqrt{\lambda}T^{\frac{1}{4}}}{\sqrt{2\pi}(1+\varepsilon)}\frac{1}{\log((\zeta-1)T+1)}\frac{1}{((\zeta-1)T+1)^{\frac{(1+\varepsilon)^2}{2D^2\lambda}\log((\zeta-1)T+1)}} \\ \leq& c_{\varepsilon,\zeta,\lambda}T^{-\beta} \end{aligned} \tag{11}$$

where $\beta = \frac{(1+\varepsilon)^2}{2D^2\lambda} - \frac{1}{4}$ and we assume $\log((\zeta-1)T+1) = \Omega(1)$ with respect to T. Together we prove the theorem 2.2. □

C GENERALIZATION

Sorscher et al. (2022) analysed the generalization of static training scheme in the teacher-student perceptron setting, where the teacher is an "oracle" generating labels. For the training set $\mathcal{T} = \{\boldsymbol{x}_i, y_i\}_{i=1}^{|\mathcal{T}|}$, assume $\boldsymbol{x}_i \sim \mathcal{N}(0, \mathbf{I})$ and there exists an oracle model $\boldsymbol{w}_o \in \mathbb{R}^d$ which generates the labels such that $y_i = \text{sign}\left(\boldsymbol{w}_o^\top \boldsymbol{x}_i\right)$ for all i. Without loss of generality, the oracle is assumed to be drawn form a sphere. Sorscher et al. (2022) works in a high dimensional statistics where $|\mathcal{T}|, d \to \infty$ but the ratio $\alpha = |\mathcal{T}|/d$ remains $\mathcal{O}(1)$.

Following the static training scheme, a lower fidelity estimator $\boldsymbol{w}_{\text{estimate}}$ which has angle θ relative to the oracle $\boldsymbol{w}_o$ is used to evaluate the candidate instances, and those with smaller classification margin $|\boldsymbol{w}_{\text{estimate}}^\top \boldsymbol{x}_i|$ along the estimator $\boldsymbol{w}_{\text{estimate}}$ are picked. The selection results in a subset $\mathcal{S}$. $\mathcal{S}$ follows $p(z)$, a truncated Gaussian distribution along $\boldsymbol{w}_{\text{estimate}}$, while the other directions are still kept isotropic. More specifically, given a keeping ratio γ, the corresponding selection margin is $\kappa = H^{-1}\left(\frac{1-\gamma}{2}\right)$ and thus the subset distribution along $\boldsymbol{w}_{estimate}$ is $p(z) = \frac{e^{-z^2/2}}{\sqrt{2\pi}\gamma}\Theta(\kappa - |z|)$, where $\Theta(x)$ is the Heaviside function and $H(x) = 1 - \Phi(x)$ where $\Phi(x)$ is the cumulative distribution function (CDF) of standard Gaussian.

The generalization error of the model trained on the subset $\mathcal{S}$ takes the form $\mathcal{E}(\alpha, \gamma, \theta)$. That is, the error is determined by γ the keeping ratio, α which indicates the abundance of training samples before selection, and θ which shows the closeness of the estimator to the oracle model. The full set of self-consistent equations characterizing $\mathcal{E}(\alpha, \gamma, \theta)$ is given as

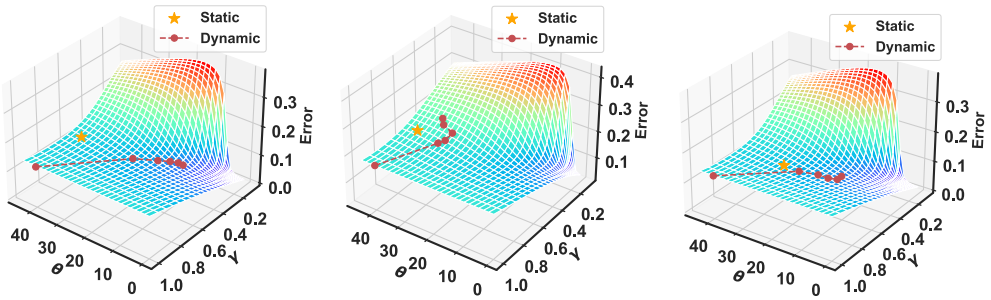
$$
\begin{aligned}
\frac{R-\rho\cos\theta}{\sin^2\theta} &= \frac{\alpha}{\pi\Lambda}\left\langle \int_{-\infty}^{\nu} d\tau \exp\left(-\frac{\Delta(\tau,z)}{2\Lambda^2}\right)(\nu-\tau)\right\rangle_z \\
1-\frac{\rho^2+R^2-2\rho R\cos\theta}{\sin^2\theta} &= 2\alpha\left\langle \int_{-\infty}^{\nu} d\tau \frac{e^{-\frac{(\tau-\rho z)^2}{2(1-\rho^2)}}}{\sqrt{2\pi}\sqrt{1-\rho^2}} H\left(\frac{\Gamma(\tau,z)}{\sqrt{1-\rho^2}\Lambda}\right)(\nu-t)^2\right\rangle_z \\
\frac{\rho-R\cos\theta}{\sin^2\theta} &= 2\alpha\left\langle \int_{-\infty}^{\nu} d\tau \frac{e^{-\frac{(\tau-\rho z)^2}{2(1-\rho^2)}}}{\sqrt{2\pi}\sqrt{1-\rho^2}} H\left(\frac{\Gamma(\tau,z)}{\sqrt{1-\rho^2}\Lambda}\right)\left(\frac{z-\rho\tau}{1-\rho^2}\right)(\nu-\tau)\right. \\
&\quad \left. + \frac{1}{2\pi\Lambda}\exp\left(-\frac{\Delta(\tau,z)}{2\Lambda^2}\right)\left(\frac{\rho R-\cos\theta}{1-\rho^2}\right)(\nu-\tau)\right\rangle_z
\end{aligned} \tag{12}
$$

Where,

$$
\begin{aligned}
\Lambda &= \sqrt{\sin^2\theta - R^2 - \rho^2 + 2\rho R\cos\theta} \\
\Gamma(t,z) &= z(\rho R-\cos\theta) - \tau(R-\rho\cos\theta) \\
\Delta(t,z) &= z^2\left(\rho^2+\cos^2\theta-2\rho R\cos\theta\right) + 2\tau z(R\cos\theta-\rho) + \tau^2\sin^2\theta
\end{aligned}
$$

τ is an auxiliary field introduced by Hubbard-Stratonovich transformation. $\langle\cdot\rangle_z$ denotes expectation on $p(z)$. By solving these equations the generalization error can be easily read off as $\mathcal{E} = \cos^{-1}(R)/\pi$, where $R = \frac{\boldsymbol{w}^\top \boldsymbol{w}_o}{\|\boldsymbol{w}\|_2 \cdot \|\boldsymbol{w}_o\|}$.

D More Results on Generalization



(a) Smaller budget ($\gamma_{\text{avg}} = 50\%$). (b) Less abundant data ($\alpha = 2.1$). (c) Better estimator ($\theta = 30°$).

Figure 6: Effects of select ratio γ_{avg}, initial data abundance α and the closeness of the estimator to the oracle model θ on the generalization.

To better understand the generalization under classification margin selection $\mathcal{E}(\alpha, \gamma, \theta)$, we provide more results to individually inspect the effect of (on average) select ratio γ_{avg}, initial data abundance α and the closeness of the estimator to the oracle mode θ. As shown in Figure 6(a), we changed γ_{avg} from 60% to 50%, thus constructing a smaller selection budget case. In Figure 6(b), we use $\alpha = 2.1$ instead of $\alpha = 3.2$ to construct a less abundant data case, where the data before selection is insufficient. In Figure 6(c), we start selecting samples using a better estimator $\theta = 30°$ instead of $\theta = 40°$. All the other hyper-parameters aside from the inspected one are kept consistent to those used Figure 2(b), that is, $\gamma_{\text{avg}} = 0.6$, $\alpha = 3.2$ and $\theta = 40°$. We see that with various γ_{avg} and θ, DynaMS outperforms its static counterpart. The abundance of initial data, however, significantly

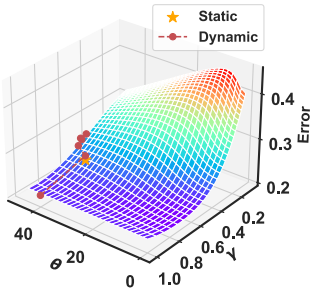


Figure 7: Generalization in a data scarce regime ($\alpha = 1.7$).

affects. When data is insufficient, data selection, both static as well as dynamic cause obvious performance degradation. Figure 7 shows a even more serious $\alpha = 1.7$, the generalization landscape is significantly changed and data selection is not recommended in this case.

E Comparison with Standard Deviation

We test each method in Table 4 and Table 5 5 times. The averaged accuracy and standard deviation are reported below in Table 6 and Table 7.

Table 6: Comparison for ResNet-18 on Cifar-10 with standard deviation.

Methods	Types	Budget	Schedule	Accuracy.
Original	-	100%	-	$95.52_{\pm 0.09}$
Random	Stat.	60%	-	$94.09_{\pm 0.23}$
EL2N_1	Stat.	60%	-	$94.55_{\pm 0.15}$
EL2N_{10}	Stat.	60%	-	$95.34_{\pm 0.13}$
GraNd_{10}	Stat.	60%	-	$95.21_{\pm 0.15}$
Forget_{10}	Stat.	60%	-	$95.29_{\pm 0.12}$
OnlineMS	OLBS.	60%	Const.	$95.21_{\pm 0.18}$
Auto-assist	OLBS.	60%	Const.	$92.37_{\pm 0.24}$
DynaRandom	Dyna.	60%	Linear	$94.45_{\pm 0.17}$
DynaCE	Dyna.	60%	Linear	$94.96_{\pm 0.21}$
Craig	Dyna.	60%	Const.	$94.36_{\pm 0.19}$
GradMatch	Dyna.	60%	Const.	$94.84_{\pm 0.17}$
DynaMS	Dyna.	60%	Linear	$95.28_{\pm 0.15}$

Table 7: Comparison for ResNet-18 and ResNet-50 on ImageNet with standard deviation

	Types	Budget	Schedule	ResNet 18		ResNet 50	
				Top1 Acc.	Top5 Acc.	Top1 Acc.	Top5 Acc.
Original	-	100%	-	$70.56_{\pm 0.04}$	$89.95_{\pm 0.02}$	$75.96_{\pm 0.04}$	$92.75_{\pm 0.02}$
Random	Stat.	60%	-	$67.16_{\pm 0.10}$	$87.50_{\pm 0.07}$	$72.46_{\pm 0.10}$	$90.85_{\pm 0.04}$
EL2N_1	Stat.	60%	-	$66.38_{\pm 0.09}$	$88.56_{\pm 0.05}$	$72.03_{\pm 0.12}$	$91.78_{\pm 0.04}$
EL2N_{10}	Stat.	60%	-	$66.46_{\pm 0.10}$	$88.73_{\pm 0.04}$	$72.18_{\pm 0.10}$	$92.02_{\pm 0.06}$
GraNd_{10}	Stat.	60%	-	$66.50_{\pm 0.06}$	$88.76_{\pm 0.03}$	$72.14_{\pm 0.06}$	$92.16_{\pm 0.03}$
Forget_{10}	Stat.	60%	-	$67.84_{\pm 0.08}$	$87.50_{\pm 0.03}$	$73.50_{\pm 0.06}$	$91.41_{\pm 0.04}$
SVP+Forget	Stat.	60%	-	-	-	$72.90_{\pm 0.10}$	$91.37_{\pm 0.04}$
SVP+Entropy	Stat.	60%	-	-	-	$73.00_{\pm 0.01}$	$91.52_{\pm 0.01}$
DynaRandom	Dyna.	60%	Power	$67.59_{\pm 0.05}$	$87.62_{\pm 0.03}$	$72.63_{\pm 0.12}$	$90.91_{\pm 0.08}$
DynaCE	Dyna.	60%	Power	$67.58_{\pm 0.10}$	$88.10_{\pm 0.04}$	$72.80_{\pm 0.08}$	$91.31_{\pm 0.03}$
Craig	Dyna.	60%	Const.	$65.32_{\pm 0.08}$	$86.92_{\pm 0.04}$	$70.69_{\pm 0.07}$	$90.72_{\pm 0.02}$
GradMatch	Dyna.	60%	Const.	$66.48_{\pm 0.11}$	$88.61_{\pm 0.04}$	$71.79_{\pm 0.07}$	$91.67_{\pm 0.03}$
DynaMS	Dyna.	60%	Linear	$68.12_{\pm 0.13}$	$88.93_{\pm 0.06}$	$74.10_{\pm 0.09}$	$92.25_{\pm 0.03}$
DynaMS	Dyna.	60%	Power	$\mathbf{68.65_{\pm 0.11}}$	$\mathbf{89.21_{\pm 0.04}}$	$\mathbf{74.56_{\pm 0.09}}$	$\mathbf{92.33_{\pm 0.03}}$
DynaMS+PSP	Dyna.	60%	Linear	-	-	$73.59_{\pm 0.09}$	$91.79_{\pm 0.07}$
DynaMS+PSP	Dyna.	60%	Power	-	-	$73.40_{\pm 0.08}$	$91.80_{\pm 0.01}$

F Implementation Details and Hyper-parameters

Subset size schedule Dynamic Selection admits more freedom in subset size schedule. In the experiments we consider the *linear schedule* and the *power schedule*. For linear schedule, the keeping ratio is determined by $\gamma_k = 1 - k \cdot a$ for $k = 1, 2, \ldots, K$, where a determines the sample reduction

ratio. γ is supposed to satisfy $\gamma_{\text{avg}} = \frac{1}{K}\sum_{k=1}^{K}\gamma_k = \gamma_s$ where γ_s is the selection ratio when a static training scheme is applied. Thus $\frac{1}{K}\sum_{k=1}^{K}|\mathcal{T}_k| = |\mathcal{S}|$, meaning the averaged number of data used in the dynamic scheme is kept equal to that of static training.

Aside from the linear scheduler, we also explore a power schedule where $\gamma_k = m \cdot k^{-r} + b$ for $k = 1, 2, \ldots, K$. Power schedule reserves more samples in late training, preventing performance degradation caused by over data pruning. Determining these hyper-parameters m, r, b is a bit tricky, we just require $\gamma_1 = 1.0$ to warm start and $\gamma_{\text{avg}} = \frac{1}{K}\sum_{k=1}^{K}\gamma_k = \gamma_s$ for fair comparison. γ_K should not be overly small, we empirically find $\gamma_K \approx \gamma - 0.1$ yield good results. For different budget $\gamma_s = \{0.6, 0.7, 0.8, 0.9\}$ the hyper-parameters are given in Appendix F, Table 8. Post process is carried out to make sure the resulting subset size sequence satisfy the above requirements.

(Killamsetty et al., 2021) utilize a *constant schedule*, where in each selection the subset size is kept constant as $\gamma_s \cdot |\mathcal{T}|$. This schedule however, do not admit selection without replacement. Linear and power schedule are all monotonically decreasing, thus are natural choices considering this. Figure 8 plots the three schedules on $\gamma_s = 0.6$ budget. In this paper we just provide a primary exploration on the subset size schedule, in depth study on the relationship between the subset size and the model performance as well as an automatic way determining the optimal subset size schedule is left for future work.

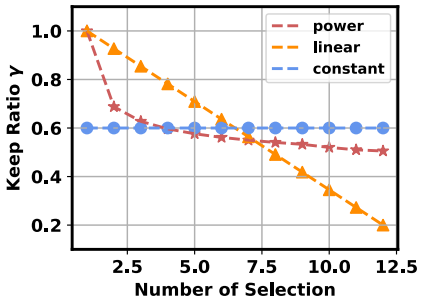


Figure 8: Different schedules for $\gamma_s = 0.6$ budget.

Hyper-parameters Finally, the detailed hyper-parameters for DynaMS on both CIFAR-10 and ImageNet datasets are shown in Table 8. Note that for DynaMS+PSP, the Max Epochs is set to be 90 on ImageNet.

Table 8: Hyper-parameters of DynaMS for different models on CIFAR-10 and ImageNet.

Hyper-parameters	CIFAR-10	ImageNet	
	ResNet-18	ResNet-18	ResNet-50
Batch Size	128	512	512
Init. Learning Rate of $\mathcal{W}$	0.1	0.1	0.1
Learning Rate Decay	Stepwise 0.2	Stepwise 0.1	Stepwise 0.1
Lr Decay milestones	{60,120,160}	{40,80}	{40,80}
Optimizer	SGD	SGD	SGD
Momentum	0.9	0.9	0.9
Nestrov	True	True	True
Weight Decay	5e-4	1e-4	1e-4
Max Epochs	200	120	120
Selection interval	10	10	10
Power Scheduler	-	60%: $m = 0.3984$, $r = 0.2371$, $b = 0.2895$ 70%: $m = 0.3476$, $r = 0.2300$, $b = 0.4275$ 80%: $m = 0.3532$, $r = 0.1349$, $b = 0.4978$ 90%: $m = 0.2176$, $r = 0.1035$, $b = 0.7078$	
Linear Scheduler	$a = 0.041$ - - -	60%: $a = 0.073$ 70%: $a = 0.055$ 80%: $a = 0.036$ 90%: $a = 0.018$	

Figure 9: Images selected at different training stages of the model. As in (Sorscher et al., 2022), we show results on ImageNet class 100 (black swan).

G VISUALIZATION OF DYNAMICALLY SELECTED IMAGES

To get a better understanding of how the selected samples look like and how they change over time, we visualize samples picked in different selection steps along the training. For $k = 1, k = 4, k = 7$ and $k = 10$, which corresponds to the 1,4,7 and 10th selection, we randomly visualize selected samples that are absent in the latter selection. E.g. the $k = 4$ row shows images picked in the 4th selection but not in the 7th selection. From Figure 9, we see that in the early selections, amounts of easy-to-recognize samples are kept. As the training proceeds, these simple images are screened out and the model focuses more on harder samples that are atypical, blurred, or with interfering objects, validating our hypothesis that samples most informative change as the model evolves. Dynamic selection is thus indispensable.

H SUMMARY OF NOTATIONS

Table 9: Summary of the notations used throughout this paper. Variables only used in theoretical analysis including the convergence analysis and the generalization analysis are grayed for better readability.

Topic	Notation	Explanation
Data (sub) Sets	$\mathcal{T}$	The full training set
	$\lvert\cdot\rvert$	Cardinality of a set
	$\boldsymbol{x}$	data sample
	y	data label
	$\mathcal{S}$	The extracted subset
	C	The number of classes
	c	The cth class
Models and Parameters	$f(\cdot)$	The model used for classification
	$\boldsymbol{w}$	Parameters of the model
	$\boldsymbol{w}^*$	Optimal model parameter
	$\boldsymbol{w}_o$	Oracle model parameter
	$\boldsymbol{W}$	Parameter of the linear classifier
	$\boldsymbol{\mathcal{W}}$	Kernel of a convolutional layers
	$\boldsymbol{g}$	gradient incurred by the model
	$\boldsymbol{g}_{\text{proxy}}$	gradient incurred by the proxy
	d	The dimension of data feature
	$h(\cdot)$	Feature extractor part of the model $f(\cdot)$
	p	Slimming factor, deciding the width of the proxy model
Selection schedule	a	Sample reduction ratio in the linear schedule
	m, r, b	Hyper-parameters controlling the power schedule
Loss Functions	$\mathcal{L}$	Generic reference to the loss function
Data Selection	$\mathcal{B}$	Decision boundary of linear classifiers
	Q	Selection interval
	M	The classification margin aka. distance of a sample to decision boundary
	γ_k	Selection budget, keep ratio of samples for the kth selection
	γ_{avg}	The averaged keep ratio of dynamic selection
	γ_s	Selection budget in static selection.
	k	Selection step
	K	The total number of selections along training
	$\mathcal{E}$	The generalization error of model trained on selected subset
	θ	Relative angle of a model to the oracle model.
	α	Aboundance of data before selection
	κ	Selection margin.
Train	t	Training epoch
	T	The total number of training epochs, $T = Q \cdot (K + 1)$
Data Distribution	Σ	Covariance of a Gaussian distribution
	λ	The largest eigenvalue of the covariance matrix
Hyper-parameters	D	Upper bound of model parameter norm
	ε, ζ, μ	Constants appear in the convergence bound.