HOW DOES SEMI-SUPERVISED LEARNING WITH PSEUDO-LABELERS WORK? A CASE STUDY

Shared by :Jiarui Jiang

January 23, 2024

Abstract

HOW DOES SEMI-SUPERVISED LEARNING WITH PSEUDO-LABELERS WORK? A CASE STUDY

Yiwen Kou[1], **Zixiang Chen**[1], **Yuan Cao**[2,3], **Quanquan Gu**[1]
[1]Department of Computer Science, University of California, Los Angeles
[2]Department of Statistics and Actuarial Science, The University of Hong Kong
[3]Department of Mathematics, The University of Hong Kong
evankou@ucla.edu, chenzx19@cs.ucla.edu, yuancao@hku.hk, qgu@cs.ucla.edu

ABSTRACT

Semi-supervised learning is a popular machine learning paradigm that utilizes a large amount of unlabeled data as well as a small amount of labeled data to facilitate learning tasks. While semi-supervised learning has achieved great success in training neural networks, its theoretical understanding remains largely open. In this paper, we aim to theoretically understand a semi-supervised learning approach based on pre-training and linear probing. In particular, the semi-supervised learning approach we consider first trains a two-layer neural network based on the unlabeled data with the help of pseudo-labelers. Then it linearly probes the pre-trained network on a small amount of labeled data. We prove that, under a certain toy data generation model and two-layer convolutional neural network, the semi-supervised learning approach can achieve nearly zero test loss, while a neural network directly trained by supervised learning on the same amount of labeled

Feature Learning

黄伟

深度学习理论爱好者

＋ 关注他

你经常看 图像处理 相关内容

继Neural Tangent Kernel (NTK)之后，深度学习理论出现了一个理论分支，人们常常称它为feature learning (theory)。不同于NTK，feature learning认为神经网络在梯度下降过程中可以学习到数据中的feature或者signal。

Feature learning理论一般会假设具体的数据生成模型，例如Gaussian mixture, signal-noise model, sparse-coding模型等，然后考察一个具体的神经网络（常常是两层网络，固定第二层权由重）在梯度下降算法下，其权重是如何学习数据中的信号和噪声。通过将复杂的神经网络的动力学转换成一个更加简单的“信号学习”和“噪声记忆”组成的动力学，feature learning theory可以刻画网络的在训练过程的优化性能以及网络收敛后的泛化能力。

由于feature learning抓住了数据（image）和神经网络动力学交互中的内在本质，其在各种算法和学习框架的可解释性上取得了空前的成功，将深度学习可解释性推向了一个新的高度。

Supervised Learning

Supervised Learning

Based on existing datasets, understand the relationship between input and output results, and then train an optimal model based on this known relationship.

examples

- classification
- regression
- decision tree
- KNN

Unsupervised Learning

Unsupervised Learning

Using a certain algorithm to train an unlabeled training set allows us to identify the underlying structure of this set of data.

examples

- K-means
- GMM
- PCA
- t-SNE

Semi-Supervised Learning

Semi-Supervised Learning

In traditional supervised learning, each training data is composed of data and labels. However, in general, only **a large amount of data can be obtained**, and **labels are difficult to obtain.** Adding labels to data requires a lot of prior knowledge, which consumes a lot of cost.

examples

- generative models
- semi-supervised support vector machines
- graph-based methods
- co-training
- consistency regularization methods
- **pseudo-labeling methods**

pseudo-labeling methods

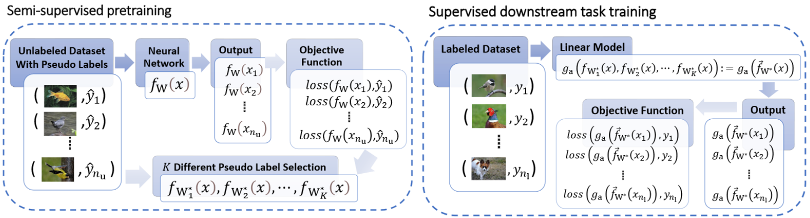

Figure 2: Illustration of our model. The left figure characterizes semi-supervised pre-train schema: NN is trained by minimizing errors between pseudo-labels $\widehat{y}$ and predictions $f_{\mathbf{W}}(\mathbf{x})$. After semi-supervised pre-training finished, the learned parameters $\{\mathbf{W}_k^*\}_{k=1}^K$ serve as pre-trained models and are adapted to a downstream task using linear probing, as shown in the right figure.

data model

Definition 3.1. Each data point $(\mathbf{x}, y)$ with $\mathbf{x} = [\mathbf{x}^{(1)\top}, \mathbf{x}^{(2)\top}]^\top \in \mathbb{R}^{2d}$ and $y \in \{-1, +1\}$ is generated as follows: the label y is generated as a Rademacher random variable; one of $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}$ is given by the feature vector $y \cdot \mathbf{v}$, the other is given by a noise vector $\boldsymbol{\xi}$ that is generated from a d-dimensional Gaussian distribution $\mathcal{N}\big(\mathbf{0}, \sigma_p^2(\mathbf{I} - \mathbf{v}\mathbf{v}^\top/\|\mathbf{v}\|_2^2)\big)$. We denote by $\mathcal{D}$ the joint distribution of $(\mathbf{x}, y)$, and denote by $\mathcal{D}_\mathbf{x}$ the marginal distribution of $\mathbf{x}$.

we consider learning a CNN with n_l labeled examples $S' = \{(x_i', y_i')\}_{i=1}^{n_l}$ generated from the distribution D and n_u labeled examples $S = \{(x_i')\}_{i=1}^{n_u}$ generated from the marginal distribution D_x

CNN model

For supervised learning, we consider a two-layer CNN whose filters are applied to the patches $\mathbf{x}^{(1)}$ and $\mathbf{x}^{(2)}$ respectively and parameters in the second layers are set to be ± 1. Then the CNN can be written as $f_{\mathbf{W}}(\mathbf{x}) = f_{\mathbf{W}}^{+1}(\mathbf{x}) - f_{\mathbf{W}}^{-1}(\mathbf{x})$ where $f_{\mathbf{W}}(\mathbf{x})^{+1}$, $f_{\mathbf{W}}(\mathbf{x})^{-1}$ are formulated as

$$\begin{aligned} f_{\mathbf{W}}^{+1}(\mathbf{x}) &= \sum_{j=1}^{m} \Big[\sigma\big(\langle \mathbf{w}_j, \mathbf{x}^{(1)}\rangle\big) + \sigma\big(\langle \mathbf{w}_j, \mathbf{x}^{(2)}\rangle\big)\Big], \\ f_{\mathbf{W}}^{-1}(\mathbf{x}) &= \sum_{j=m+1}^{2m} \Big[\sigma\big(\langle \mathbf{w}_j, \mathbf{x}^{(1)}\rangle\big) + \sigma\big(\langle \mathbf{w}_j, \mathbf{x}^{(2)}\rangle\big)\Big]. \end{aligned} \tag{3.1}$$

Here σ is activation function $\mathrm{ReLU}^q(\cdot) = [\,\cdot\,]_+^q (q > 2)$, m is the width of the network, $\mathbf{w}_j \in \mathbb{R}^d$ denotes the j-th filter, and $\mathbf{W}$ is the collection of all filters $\{\mathbf{w}_j\}_{j=1}^{2m}$. Given labeled training dataset $S' = \{(\mathbf{x}'_i, y'_i)\}_{i=1}^{n_1}$, we train the CNN model by minimizing the empirical cross-entropy loss

$$L_{S'}(\mathbf{W}) = \frac{1}{n_1} \sum_{i=1}^{n_1} L_i(\mathbf{W}),$$

where $L_i(\mathbf{W}) = \ell\big(y'_i \cdot f_{\mathbf{W}}(\mathbf{x}'_i)\big)$ with $\ell(z) = \log(1 + \exp(-z))$ denotes the individual loss for the training example $(\mathbf{x}_i, y_i)$. We minimize the empirical function $L_{S'}(\mathbf{W})$ with gradient descent as follows

$$\mathbf{w}_j^{(t+1)} = \mathbf{w}_j^{(t)} - \eta \cdot \nabla_{\mathbf{w}_j} L_{S'}(\mathbf{W}^{(t)}), \quad \mathbf{w}_j^{(0)} \sim \mathcal{N}(\mathbf{0}, \sigma_0^2 \mathbf{I}), \quad j \in [2m],$$

where $\eta > 0$ is the learning rate and σ_0 defines the scale of random initialization.

Semi-Supervised Learning Models

For semi-supervised pre-training, we assume that we have access to K pseudo-labelers $\{f_k^{\mathrm{w}}\}_{k=1}^K$. The accuracy of k-th pseudo-labeler is $p_k \in (1/2, 1)$. Then we use K pseudo-labelers to generate K pseudo-labeled dataset $\{S_k\}_{k=1}^K$, where $S_k := \left\{(\mathbf{x}_i, \widehat{y}_{k,i}) \,\middle|\, \widehat{y}_{k,i} = f_k^{\mathrm{w}}(\mathbf{x}_i)\right\}_{i=1}^{n_{\mathrm{u}}}$. Next we solve K pre-training tasks with two-layer CNN models $\{f_{\mathbf{W}_k}\}_{k=1}^K$ defined in (3.1) using $\{S_k\}_{k=1}^K$ respectively. Note that our result can cover $K = 1$ as a special case, where there is only one pseudo-labeler.

We consider learning the model parameter $\mathbf{W}_k$ by optimizing the empirical loss of both pseudo-labeled dataset S_k and labeled dataset $S' = \{(\mathbf{x}_i', y_i')\}_{i=1}^{n_{\mathrm{l}}}$ with weight decay regularization

$$L_{S_k \cup S'}(\mathbf{W}_k) = \frac{1}{n_{\mathrm{u}} + n_{\mathrm{l}}} \left(\sum_{i=1}^{n_{\mathrm{u}}} L_i(\mathbf{W}_k) + \sum_{i'=1}^{n_{\mathrm{l}}} L_{i'}(\mathbf{W}_k) \right) + \frac{\lambda}{2} \|\mathbf{W}_k\|_F^2,$$

$$\mathbf{w}_{k,j}^{(t+1)} = \mathbf{w}_{k,j}^{(t)} - \eta \cdot \nabla_{\mathbf{w}_{k,j}} L_{S_k \cup S'}(\mathbf{W}_k^{(t)}), \quad \mathbf{w}_{k,j}^{(0)} \sim \mathcal{N}(\mathbf{0}, \sigma_0^2 \mathbf{I}_d), \quad j \in [2m], k \in [K]$$

Downstream Task: Linear Model

$$g_{\mathbf{a}}(\mathbf{x}) = \sum_{k=1}^{K} a_k f_{\mathbf{W}_k^*}(\mathbf{x}),$$

$$L_{S'}(\mathbf{a}) = \frac{1}{n} \sum_{i=1}^{n} \ell\left(y_i' \cdot g_{\mathbf{a}}(\mathbf{x}_i')\right).$$

$$\mathbf{a}^{(t+1)} = \mathbf{a}^{(t)} - \eta \cdot \nabla_{\mathbf{a}} L_{S'}(\mathbf{a}^{(t)}),\ \mathbf{a}^{(0)} = \mathbf{0}.$$

Main Results

Condition 4.1. The strength of the signal is $\|\mathbf{v}\|_2^2 = \Theta(d)$, the noise variance is $\sigma_p = \Theta(d^\epsilon)$, where $0 < \epsilon < 1/8$ is a small constant, and the width of the network satisfies $m = \mathrm{polylog}(d)$. We also assume that the size of the unlabeled dataset $n_\mathrm{u} = \Omega(d^{4\epsilon})$, and labeled data $n_\mathrm{l} = \widetilde{\Theta}(1)$. For both supervise learning and semi-supervised learning settings, we initialize the weight with $\sigma_0 = \Theta(d^{-3/4})$. For semi-supervised learning, we require $\lambda = o(d^{3/4})$ and assume that there exists a constant C such that for all pseudo-labelers, their test accuracy $p_k > 1/2 + C$.

Since we generate the noise patch from the Gaussian distribution, the strength of the noise patch is $\|\boldsymbol{\xi}\|_2^2 \approx d^{1+\epsilon}$ by standard concentration inequalities, which is larger than the strength of the signal patch $\|\mathbf{v}\|_2^2 = \Theta(d)$. Therefore, Condition 4.1 defines a setting with large noises. The condition of $d \gg n_u \gg n_l$ further ensures that learning is in a sufficiently over-parameterized setting. Here we only require the neural network width m to be polylogarithmic in the dimension d and require the psudolablers to perform better than a random guess.

Main Results

Theorem 4.2 (Semi-supervised Learning: Pre-training)**.** Let $k \in [K]$ and consider the semi-supervised pre-training of $f_{\mathbf{W}_k}(\mathbf{x})$. For any test data point $(\mathbf{x}, y)$, denote $\widehat{y} = f_k^{\mathrm{w}}(\mathbf{x})$. Then under Condition 4.1, after $T_0 = \widetilde{\Theta}(d^{-\frac{3}{4}}\eta^{-1})$ training iterations with learning rate $\eta = O(d^{-1.1})$, the trained neural network can achieve nearly 0 test error on the distribution $\mathcal{D}$: $\mathbb{P}_{(\mathbf{x},y)\sim\mathcal{D}}[y \cdot f_{\mathbf{W}_k^{(T_0)}}(\mathbf{x}) \leq 0] = o(1)$.

Theorem 4.2 characterizes the prediction power of the feature representation learned in the pre-trained models using unlabeled data. For any test data point $(\mathbf{x}, y)$, the sign of y can be predicted based on $f_{\mathbf{W}^{(T_0)}}(\mathbf{x})$ with high probability.

Main Results

Theorem 4.3 (Semi-supervised Learning: Downstream). Let $\{f_{\mathbf{W}_k^{(T_0^k)}}\}_{k=1}^{a}$ be the neural networks trained according to the K pre-training tasks, and consider the learning of the downstream task based in $\{f_{\mathbf{W}_k^{(T_0^k)}}\}_{k=1}^{d}$. Under Condition 4.1, after $T' = \Theta(d^{0.1}/\eta)$ iterations with learning rate $\eta = \Theta(1)$, with probability $1 - o(1)$, the obtained $\mathbf{a}^{(T')}$ satisfies:

- Training error is 0: $\frac{1}{n}\sum_{i=1}^{n} \mathbb{1}[y_i \cdot g_{\mathbf{a}^{(T')}}(\mathbf{x}_i) \leq 0] = 0$.
- Test error and loss are nearly 0: $\mathbb{P}_{(\mathbf{x},y)\sim\mathcal{D}}[y \cdot g_{\mathbf{a}^{(T')}}(\mathbf{x}) \leq 0] = o(1)$, $L_{\mathcal{D}}\big(\mathbf{a}^{(T')}\big) = o(1)$.

Theorem 4.3 shows that the feature representation learned based on the semi-supervised pre-training can ensure small training and test errors for the supervised downstream task. Notably, this result holds even though we assume that there are only a constant number of labeled data. This shows that semi-supervised learning can significantly reduce the need for a large labeled training dataset. For comparison, we also have the following guarantees on the performance of standard supervised learning of CNNs.

Main Results

Theorem 4.4 (Supervised Learning)**.** Under supervised learning setting, after gradient descent for $T = \widetilde{\Theta}(d^{(1/4-\epsilon)q-3/2}\eta^{-1})$ iterations with learning rate $\eta = O(d^{-1-2\epsilon})$, then there exists $t \leq T$ such that with probability $1 - o(1)$ the CNNs defined in (3.1) with parameter $\mathbf{W}^{(t)}$ satisfies:

- Training loss is nearly zero: $L_{S'}\big(\mathbf{W}^{(t)}\big) = o(1)$.
- Test loss is high: $L_{\mathcal{D}}\big(\mathbf{W}^{(t)}\big) = \Theta(1)$.

	Semi-supervised		Supervised
	Pre-train	Downstream	
Training error	0.1753±0.0259	0	0
Test error	0	0	0.4982± 0.0208
Training loss	0.4155±0.0418	0.0150±0.0022	$(6.473\pm5.031)\times10^{-7}$
Test loss	0.2200±0.0886	0.0182±0.0021	0.6931±0.0005

Table 1: Training error and loss, test error and loss for semi-supervised and supervised learning.

Proof Sketch

Figure: some definitions

Our study of the pre-training focuses on two aspects of the training process: *feature learning* and *noise memorization*. Specifically, we aim to monitor how the filters in the CNN model learn the feature vector $\mathbf{v}$ and the noise vectors $\boldsymbol{\xi}_i$'s. Therefore, we introduce the following notations.

$$\widehat{\Lambda}_1^{(t)} := \max_{1 \le j \le m} \langle \mathbf{w}_j^{(t)}, \mathbf{v} \rangle, \ \bar{\Lambda}_1^{(t)} := \max_{1 \le j \le m} -\langle \mathbf{w}_j^{(t)}, \mathbf{v} \rangle,$$

$$\widehat{\Lambda}_{-1}^{(t)} := \max_{m+1 \le j \le 2m} -\langle \mathbf{w}_j^{(t)}, \mathbf{v} \rangle, \ \bar{\Lambda}_{-1}^{(t)} := \max_{m+1 \le j \le 2m} \langle \mathbf{w}_j^{(t)}, \mathbf{v} \rangle,$$

$$\Gamma_i^{(t)} := \max_{1 \le j \le 2m} \langle \mathbf{w}_j^{(t)}, \boldsymbol{\xi}_i \rangle, \ \Gamma_i'^{(t)} := \max_{1 \le j \le 2m} \langle \mathbf{w}_j^{(t)}, \boldsymbol{\xi}_i' \rangle, \ \Gamma^{(t)} = \max \left\{ \max_{i \in [n_u]} \Gamma_i^{(t)}, \max_{i \in [n_l]} \Gamma_i'^{(t)}, \right\}.$$

the larger $\hat{\Lambda}$, the better
the smaller $\bar{\Lambda}$, the better
the smaller Γ, the better

Proof Sketch

Lemma 5.1. Assume we use both unlabeled data with pseudo-labels generated by the pseudo-labeler and labeled data for the training of our CNN model. Then for $r \in \{\pm 1\}$, let T_r be the first iteration that $\widehat{\Lambda}_r^{(t)}$ reaches $\Theta(1/m)$, then for $t \in [0, T_r]$, we have

$$\widehat{\Lambda}_r^{(t+1)} \geq (1-\eta\lambda) \cdot \widehat{\Lambda}_r^{(t)} + \eta \cdot C \cdot \Theta(d) \cdot (\widehat{\Lambda}_r^{(t)})^{q-1}, r \in \{\pm 1\},$$
$$\bar{\Lambda}_r^{(t+1)} \leq (1-\eta\lambda) \cdot \bar{\Lambda}_r^{(t)}, r \in \{\pm 1\},$$
$$\Gamma^{(t+1)} \leq (1-\eta\lambda) \cdot \Gamma^{(t)} + \eta \cdot \widetilde{\Theta}(d^{1-2\epsilon}) \cdot (\Gamma^{(t)})^{q-1},$$

Lemma 5.2. Assume we use only labeled data for the training of our CNN model. Then for $i \in [n_l]$, let T_i' be the first iteration that $\Gamma_i'^{(t)}$ reaches $\Theta(1/m)$, then we have

$$\widehat{\Lambda}_r^{(t+1)} \leq (1-\eta\lambda) \cdot \widehat{\Lambda}_r^{(t)} + \eta \cdot \Theta(d) \cdot \left((\widehat{\Lambda}_r^{(t)})^{q-1} + (\bar{\Lambda}_r^{(t)})^{q-1}\right), r \in \{\pm 1\},$$
$$\bar{\Lambda}_r^{(t+1)} \leq (1-\eta\lambda) \cdot \bar{\Lambda}_r^{(t)}, r \in \{\pm 1\},$$
$$\Gamma_i'^{(t+1)} \geq (1-\eta\lambda) \cdot \Gamma_i'^{(t)} + \eta \cdot \widetilde{\Theta}(d^{1+2\epsilon}) \cdot (\Gamma_i'^{(t)})^{q-1}, i \in [n_l], \text{ for } t \in [0, T_i'].$$

Proof Sketch

Lemma 5.3. If both pseudo-labeled and labeled data are used to train CNN, for $r \in \{\pm 1\}$, let T_r be the first iteration that $\widehat{\Lambda}_r^{(t)}$ reaches $\Theta(1/m)$ respectively. Let $T_0 = \max_{r\in\{\pm 1\}}\{T_r\}$. Then, it holds that $\widehat{\Lambda}_r^{(T_0)} = \widetilde{\Theta}(1)$, $\bar{\Lambda}_r^{(t)} = \widetilde{O}(d^{-\frac{1}{4}})$ and $\Gamma^{(t)} = \widetilde{O}(d^{-\frac{1}{4}+\epsilon})$ for all $t \in [0, T_0]$.

Lemma 5.4. If only labeled data are used to train CNN, for $i \in [n_1]$, let T'_i be the first iteration that $\Gamma_i'^{(t)}$ reaches $\Theta(1/m)$. Let $T'_0 = \max_{i\in[n_1]} T'_i$. Then, it holds that $\widehat{\Lambda}_r = \widetilde{O}(d^{-\frac{1}{4}})$, $\bar{\Lambda}_r = \widetilde{O}(d^{-\frac{1}{4}})$ for $r \in \{\pm 1\}$ and $\Gamma_i'^{(t)} = \widetilde{\Theta}(1)$ for $i \in [n_1]$.

Lemma 5.5. For any learning rate $\eta = \Theta(1)$, we have $\left\|\mathbf{a}^{(t)}\right\|_1 = \log(t)/\widetilde{\Theta}(1)$. For any labeled data $(\mathbf{x}'_i, y'_i) \in S'$, we have with high probability that $y'_i \cdot f_{\mathbf{W}^{(t)}}(\mathbf{x}'_i) = \left\|\mathbf{a}^{(t)}\right\|_1 \cdot \widetilde{\Theta}(1)$. For any newly generated data $(\mathbf{x}, y) \sim \mathcal{D}$, we also have with high probability that $y \cdot f_{\mathbf{W}^{(t)}}(\mathbf{x}) = \left\|\mathbf{a}^{(t)}\right\|_1 \cdot \widetilde{\Theta}(1)$.

With the help of the above lemma and note that training error and test error are related to $y \cdot f_{\mathbf{W}^{(T_0)}}(\mathbf{x})$ and test loss is related to $\|\mathbf{a}^{(T_0)}\|_1$, we can prove that after $T = \Theta(d^{0.1}/\eta)$ iterations with learning rate $\eta = \Theta(1)$, the model can achieve nearly zero training error, test error, training loss and test loss.

Gradient

Lemma A.1 (Gradient Calculation)**.** The gradient of loss function $L_S(\mathbf{W})$ with respect to weight parameters $\mathbf{w}_j$ is

$$\nabla_{\mathbf{w}_j} L_{S\cup S'}(\mathbf{W}) = -\frac{q}{n_{\mathrm{l}} + n_{\mathrm{u}}} \Bigg(\sum_{i=1}^{n_{\mathrm{u}}} c_i \widehat{y}_i \big([\langle \mathbf{w}_j, y_i \cdot \mathbf{v} \rangle]_+^{q-1} \cdot y_i \cdot \mathbf{v} + [\langle \mathbf{w}_j, \boldsymbol{\xi}_i \rangle]_+^{q-1} \cdot \boldsymbol{\xi}_i \big)$$
$$+ \sum_{i=1}^{n_{\mathrm{l}}} b_i y_i' \big([\langle \mathbf{w}_j, y_i' \cdot \mathbf{v} \rangle]_+^{q-1} \cdot y_i' \cdot \mathbf{v} + [\langle \mathbf{w}_j, \boldsymbol{\xi}_i' \rangle]_+^{q-1} \cdot \boldsymbol{\xi}_i' \big) \Bigg) + \lambda \cdot \mathbf{w}_j,$$

for $1 \le j \le m$; and

$$\nabla_{\mathbf{w}_j} L_{S\cup S'}(\mathbf{W}) = \frac{q}{n_{\mathrm{l}} + n_{\mathrm{u}}} \Bigg(\sum_{i=1}^{n_{\mathrm{u}}} c_i \widehat{y}_i \big([\langle \mathbf{w}_j, y_i \cdot \mathbf{v} \rangle]_+^{q-1} \cdot y_i \cdot \mathbf{v} + [\langle \mathbf{w}_j, \boldsymbol{\xi}_i \rangle]_+^{q-1} \cdot \boldsymbol{\xi}_i \big)$$
$$+ \sum_{i=1}^{n_{\mathrm{l}}} b_i y_i' \big([\langle \mathbf{w}_j, y_i' \cdot \mathbf{v} \rangle]_+^{q-1} \cdot y_i' \cdot \mathbf{v} + [\langle \mathbf{w}_j, \boldsymbol{\xi}_i' \rangle]_+^{q-1} \cdot \boldsymbol{\xi}_i' \big) \Bigg) + \lambda \cdot \mathbf{w}_j,$$

for $m + 1 \le j \le 2m$, where $-\ell'\big(\widehat{y}_i \cdot f_{\mathbf{W}}(\mathbf{x}_i)\big) = \exp\left[-\widehat{y}_i \cdot f_{\mathbf{W}}(\mathbf{x}_i)\right]/(1 + \exp\left[-\widehat{y}_i \cdot f_{\mathbf{W}}(\mathbf{x}_i)\right])$ is denoted by c_i and $-\ell'(y_i' \cdot f_{\mathbf{W}}(\mathbf{x}_i')) = \exp[-y_i' \cdot f_{\mathbf{W}}(\mathbf{x}_i')]/(1 + \exp[-y_i' \cdot f_{\mathbf{W}}(\mathbf{x}_i')])$ is denoted by b_i.

Inner Product Update Rule

Lemma A.2 (Inner Product Update Rule)**.** The feature learning and noise memorization performance of gradient descent can be formulated by

$$\langle \mathbf{w}_j^{(t+1)}, \mathbf{v}\rangle = (1-\eta\lambda)\cdot\langle \mathbf{w}_j^{(t)}, \mathbf{v}\rangle + \frac{q\eta u_j}{n_l + n_u}\Bigg(\sum_{i=1}^{n_u} y_i \widehat{y}_i c_i^{(t)} [\langle \mathbf{w}_j^{(t)}, y_i \cdot \mathbf{v}\rangle]_+^{q-1} \|\mathbf{v}\|_2^2 + \sum_{i=1}^{n_l} b_i^{(t)} [\langle \mathbf{w}_j^{(t)}, y_i' \cdot \mathbf{v}\rangle]_+^{q-1} \|\mathbf{v}\|_2^2\Bigg),$$

$$\langle \mathbf{w}_j^{(t+1)}, \boldsymbol{\xi}_l\rangle = (1-\eta\lambda)\cdot\langle \mathbf{w}_j^{(t)}, \boldsymbol{\xi}_l\rangle + \frac{q\eta u_j}{n_l + n_u}\Bigg(\sum_{i=1}^{n_u} \widehat{y}_i c_i^{(t)} [\langle \mathbf{w}_j^{(t)}, \boldsymbol{\xi}_i\rangle]_+^{q-1} \langle \boldsymbol{\xi}_i, \boldsymbol{\xi}_l\rangle + \sum_{i=1}^{n_l} y_i' b_i^{(t)} [\langle \mathbf{w}_j^{(t)}, \boldsymbol{\xi}_i'\rangle]_+^{q-1} \langle \boldsymbol{\xi}_i', \boldsymbol{\xi}_l\rangle\Bigg),$$

$$\langle \mathbf{w}_j^{(t+1)}, \boldsymbol{\xi}_l'\rangle = (1-\eta\lambda)\cdot\langle \mathbf{w}_j^{(t)}, \boldsymbol{\xi}_l'\rangle + \frac{q\eta u_j}{n_l + n_u}\Bigg(\sum_{i=1}^{n_u} \widehat{y}_i c_i^{(t)} [\langle \mathbf{w}_j^{(t)}, \boldsymbol{\xi}_i\rangle]_+^{q-1} \langle \boldsymbol{\xi}_i, \boldsymbol{\xi}_l'\rangle + \sum_{i=1}^{n_l} y_i' b_i^{(t)} [\langle \mathbf{w}_j^{(t)}, \boldsymbol{\xi}_i'\rangle]_+^{q-1} \langle \boldsymbol{\xi}_i', \boldsymbol{\xi}_l'\rangle\Bigg),$$

where $j \in [2m]$, $l \in [n_u]$ and $u_j := \mathbb{1}_{[1\le j\le m]} - \mathbb{1}_{[m+1\le j\le 2m]}$.

Some Interesting Lemma

Lemma A.5. As long as $\max_{r\in\{\pm 1\}}\{\widehat{\Lambda}_r^{(t)}, \bar{\Lambda}_r^{(t)}\} \le \Theta(m^{-1})$, we have $c_i^{(t)} := -\ell'\big(\widehat{y}_i \cdot f_{\mathbf{W}^{(t)}}(\mathbf{x}_i)\big)$ and $b_i^{(t)} := -\ell'\big(y_i' \cdot f_{\mathbf{W}^{(t)}}(\mathbf{x}_i')\big)$ remains $1/2 \pm o(1)$.

Lemma A.6. For any $\delta < 1/2$, with probability at least $1 - 2\delta$ over pseudo-labels generated by the pseudo-labeler, we have

$$\left|\frac{1}{n_{\mathrm{u}}}\sum_{i=1}^{n_{\mathrm{u}}} \widehat{y}_i y_i c_i^{(t)} - \Big(p - \frac{1}{2}\Big)\right| < \sqrt{\frac{1}{8n_{\mathrm{u}}}\log\frac{1}{\delta}} + o(1),$$

where $o(1)$ is with respect to d.

Lemma A.9. For $\widehat{\Lambda}_1^{(t)} := \max_{1\le j\le m}\langle \mathbf{w}_j^{(t)}, \mathbf{v}\rangle$ and $\widehat{\Lambda}_{-1}^{(t)} := \max_{m+1\le j\le 2m}\langle \mathbf{w}_j^{(t)}, -\mathbf{v}\rangle$, we have with high probability that

$$\widehat{\Lambda}_r^{(t+1)} \ge (1-\eta\lambda)\cdot\widehat{\Lambda}_r^{(t)} + \eta\cdot\left(p-\frac{1}{2}\right)\cdot\Theta(d)\cdot(\widehat{\Lambda}_r^{(t)})^{q-1}, r\in\{\pm 1\}.$$

For $\bar{\Lambda}_1^{(t)} := \max_{m+1\le j\le 2m}\langle \mathbf{w}_j^{(t)}, \mathbf{v}\rangle$ and $\bar{\Lambda}_1^{(t)} := \max_{1\le j\le m}\langle \mathbf{w}_j^{(t)}, -\mathbf{v}\rangle$, we have with high probability that

$$\bar{\Lambda}_r^{(t+1)} \le (1-\eta\lambda)\cdot\bar{\Lambda}_r^{(t)}, r\in\{\pm 1\}.$$

Proof of Lemma A.9. We first prove the former inequality. Let $j^* = \arg\max_{1\le j\le m}\langle \mathbf{w}_j^{(t)}, \mathbf{v}\rangle$ and note that $u_{j^*} = \mathbb{1}_{[1\le j\le m]} - \mathbb{1}_{[m+1\le j\le 2m]} = 1$, then we have

$$\widehat{\Lambda}_1^{(t+1)} \ge \langle \mathbf{w}_{j^*}^{(t+1)}, \mathbf{v}\rangle$$

$$= (1-\eta\lambda)\cdot\langle \mathbf{w}_{j^*}^{(t)}, \mathbf{v}\rangle + \frac{q\eta}{n_l+n_u}\Bigg(\underbrace{\sum_{i=1}^{n_u} y_i\widehat{y}_i c_i^{(t)}[\langle \mathbf{w}_{j^*}^{(t)}, y_i\cdot\mathbf{v}\rangle]_+^{q-1}\|\mathbf{v}\|_2^2}_{\clubsuit} + \underbrace{\sum_{i=1}^{n_l} b_i^{(t)}[\langle \mathbf{w}_{j^*}^{(t)}, y_i'\cdot\mathbf{v}\rangle]_+^{q-1}\|\mathbf{v}\|_2^2}_{\bigstar}\Bigg)$$

Then we respectively estimate terms $\clubsuit$ and $\bigstar$.

Calculate ♣

For ♣, note the definition of j^* that $\widehat{\Lambda}_1^{(t)} = \langle \mathbf{w}_{j^*}^{(t)}, \mathbf{v} \rangle$ and note the increasing property of $\widehat{\Lambda}_1^{(t)}$ and $\widehat{\Lambda}_1^{(0)} > 0$ with high probability, we have $\langle \mathbf{w}_{j^*}^{(t)}, \mathbf{v} \rangle > 0$. It follows that

$$\begin{aligned}
\underbrace{\sum_{i=1}^{n_u} y_i \widehat{y}_i c_i^{(t)} [\langle \mathbf{w}_{j^*}^{(t)}, y_i \cdot \mathbf{v} \rangle]_+^{q-1} \|\mathbf{v}\|_2^2}_{\clubsuit} &= \sum_{i \in S_1} y_i \widehat{y}_i c_i^{(t)} [\langle \mathbf{w}_{j^*}^{(t)}, \mathbf{v} \rangle]_+^{q-1} \|\mathbf{v}\|_2^2 + \sum_{i \in S_{-1}} y_i \widehat{y}_i c_i^{(t)} [-\langle \mathbf{w}_{j^*}^{(t)}, \mathbf{v} \rangle]_+^{q-1} \|\mathbf{v}\|_2^2 \\
&= \sum_{i \in S_1} y_i \widehat{y}_i c_i^{(t)} [\langle \mathbf{w}_{j^*}^{(t)}, \mathbf{v} \rangle]_+^{q-1} \|\mathbf{v}\|_2^2 \\
&= \left(\sum_{i \in S_1} y_i \widehat{y}_i c_i^{(t)} \right) \cdot \|\mathbf{v}\|_2^2 \cdot (\widehat{\Lambda}_1^{(t)})^{q-1} \\
&= n_1 \cdot \left(p - \frac{1}{2} \pm o(1) \right) \cdot \|\mathbf{v}\|_2^2 \cdot (\widehat{\Lambda}_1^{(t)})^{q-1}, \qquad \text{(A.7)}
\end{aligned}$$

where $S_1 := \{(\mathbf{x}_i, y_i) | y_i = 1, i \in [n_u]\}$, $S_{-1} := \{(\mathbf{x}_i, y_i) | y_i = -1, i \in [n_u]\}$, $n_1 = |S_1|$ and the last equality is due to (A.6).

Calculate ★

For ★, similarly we have

$$\underbrace{\sum_{i=1}^{n_1} b_i^{(t)}[\langle \mathbf{w}_{j^*}^{(t)}, y_i' \cdot \mathbf{v}\rangle]_+^{q-1}\|\mathbf{v}\|_2^2}_{\star} = \sum_{i\in S_1'} b_i^{(t)}[\langle \mathbf{w}_{j^*}^{(t)}, \mathbf{v}\rangle]_+^{q-1}\|\mathbf{v}\|_2^2 + \sum_{i\in S_{-1}'} b_i^{(t)}[-\langle \mathbf{w}_{j^*}^{(t)}, \mathbf{v}\rangle]_+^{q-1}\|\mathbf{v}\|_2^2$$

$$= \sum_{i\in S_1'} b_i^{(t)}[\langle \mathbf{w}_{j^*}^{(t)}, \mathbf{v}\rangle]_+^{q-1}\|\mathbf{v}\|_2^2$$

$$= \Big(\sum_{i\in S_1'} b_i^{(t)}\Big) \cdot \|\mathbf{v}\|_2^2 \cdot (\widehat{\Lambda}_1^{(t)})^{q-1}$$

$$= n_1' \cdot \Big(\frac{1}{2} \pm o(1)\Big) \cdot \|\mathbf{v}\|_2^2 \cdot (\widehat{\Lambda}_1^{(t)})^{q-1}, \tag{A.8}$$

Plug ♣ and ★

$$
\begin{aligned}
&\widehat{\Lambda}_1^{(t+1)} \\
&\geq (1-\eta\lambda)\cdot\widehat{\Lambda}_1^{(t)} + \frac{q\eta}{n_l+n_u}\left(n_1\cdot\left(p-\frac{1}{2}\pm o(1)\right)\cdot\|\mathbf{v}\|_2^2\cdot\left(\widehat{\Lambda}_1^{(t)}\right)^{q-1} + n_1'\cdot\left(\frac{1}{2}\pm o(1)\right)\cdot\|\mathbf{v}\|_2^2\cdot\left(\widehat{\Lambda}_1^{(t)}\right)^{q-1}\right) \\
&= (1-\eta\lambda)\cdot\widehat{\Lambda}_1^{(t)} + \frac{q\eta n_1}{n_l+n_u}\cdot\left(p-\frac{1}{2}\pm o(1)\right)\cdot\|\mathbf{v}\|_2^2\cdot\left(\widehat{\Lambda}_1^{(t)}\right)^{q-1} + \frac{q\eta n_1'}{n_l+n_u}\cdot\left(\frac{1}{2}\pm o(1)\right)\cdot\|\mathbf{v}\|_2^2\cdot\left(\widehat{\Lambda}_1^{(t)}\right)^{q-1} \\
&= (1-\eta\lambda)\cdot\widehat{\Lambda}_1^{(t)} + q\eta\cdot\left(\frac{n_1}{n_l+n_u}\cdot\left(p-\frac{1}{2}\pm o(1)\right)+\frac{n_1'}{n_l+n_u}\cdot\left(\frac{1}{2}\pm o(1)\right)\right)\cdot\|\mathbf{v}\|_2^2\cdot\left(\widehat{\Lambda}_1^{(t)}\right)^{q-1} \\
&= (1-\eta\lambda)\cdot\widehat{\Lambda}_1^{(t)} + q\eta\cdot\left(\underbrace{\frac{n_1}{n_l+n_u}\cdot\left(p-\frac{1}{2}\right)+\frac{n_1'}{n_l+n_u}\cdot\frac{1}{2}}_{\spadesuit}\pm o(1)\right)\cdot\|\mathbf{v}\|_2^2\cdot\left(\widehat{\Lambda}_1^{(t)}\right)^{q-1}. \quad \text{(A.9)}
\end{aligned}
$$

$$
\begin{aligned}
\underbrace{\frac{n_1}{n_l+n_u}\cdot\left(p-\frac{1}{2}\right)+\frac{n_1'}{n_l+n_u}\cdot\frac{1}{2}}_{\spadesuit} &= \frac{n_u}{2(n_l+n_u)}\cdot\left(p-\frac{1}{2}\right)+\frac{n_l}{2(n_l+n_u)}\cdot\frac{1}{2}\pm o(1) \\
&= \frac{1}{2}\cdot\left(p-\frac{1}{2}\right)\pm o(1) \qquad \text{(A.10)}
\end{aligned}
$$

$$
\begin{aligned}
\widehat{\Lambda}_1^{(t+1)} &\geq (1-\eta\lambda)\cdot\widehat{\Lambda}_1^{(t)} + q\eta\cdot\left(\frac{1}{2}\cdot\left(p-\frac{1}{2}\right)\pm o(1)\right)\cdot\|\mathbf{v}\|_2^2\cdot\left(\widehat{\Lambda}_1^{(t)}\right)^{q-1} \\
&= (1-\eta\lambda)\cdot\widehat{\Lambda}_1^{(t)} + \eta\cdot\left(p-\frac{1}{2}\right)\cdot\Theta(d)\cdot\left(\widehat{\Lambda}_1^{(t)}\right)^{q-1},
\end{aligned}
$$

Feature Learning Part.2

$$
\begin{aligned}
\bar{\Lambda}_1^{(t+1)} &= \langle \mathbf{w}_{j\natural}^{(t+1)}, \mathbf{v} \rangle \\
&= (1-\eta\lambda) \cdot \langle \mathbf{w}_{j\natural}^{(t)}, \mathbf{v} \rangle - \frac{q\eta}{n_{\mathrm{l}} + n_{\mathrm{u}}} \Bigg(\underbrace{\sum_{i=1}^{n_{\mathrm{u}}} y_i \widehat{y}_i c_i^{(t)} [\langle \mathbf{w}_{j\natural}^{(t)}, y_i \cdot \mathbf{v} \rangle]_+^{q-1} \|\mathbf{v}\|_2^2}_{\clubsuit} \\
&\quad + \underbrace{\sum_{i=1}^{n_{\mathrm{l}}} b_i^{(t)} [\langle \mathbf{w}_{j\natural}^{(t)}, y_i' \cdot \mathbf{v} \rangle]_+^{q-1} \|\mathbf{v}\|_2^2}_{\bigstar} \Bigg).
\end{aligned}
$$

here we have $\clubsuit \geq 0$ and $\bigstar \geq 0$ so:

$$
\bar{\Lambda}_1^{(t+1)} \leq (1-\eta\lambda) \cdot \langle \mathbf{w}_{j\natural}^{(t)}, \mathbf{v} \rangle \leq (1-\eta\lambda) \bar{\Lambda}_1^{(t)}.
$$

Noise Memorization

Lemma A.11. For $\Gamma_i^{(t)} := \max_{j\in[2m]}\langle \mathbf{w}_j, \boldsymbol{\xi}_i\rangle, i \in [n_{\mathrm{u}}]$, $\Gamma_i'^{(t)} := \max_{j\in[2m]}\langle \mathbf{w}_j, \boldsymbol{\xi}_i'\rangle, i \in [n_{\mathrm{l}}]$, $\Gamma^{(t)} := \max\{\max_{i\in[n_{\mathrm{u}}]}\Gamma_i^{(t)}, \max_{i\in[n_{\mathrm{l}}]}\Gamma_i'^{(t)}\}$, we have with high probability that

$$\Gamma_i^{(t+1)} \le (1-\eta\lambda)\cdot\Gamma_i^{(t)} + \eta\cdot\max\left\{\widetilde{\Theta}(d^{\frac{1}{2}+2\epsilon}), \widetilde{\Theta}\Big(\frac{d^{1+2\epsilon}}{n_{\mathrm{u}}}\Big)\right\}\cdot\big(\Gamma^{(t)}\big)^{q-1}, i \in [n_{\mathrm{l}}],$$

$$\Gamma_i'^{(t+1)} \le (1-\eta\lambda)\cdot\Gamma_i'^{(t)} + \eta\cdot\max\left\{\widetilde{\Theta}(d^{\frac{1}{2}+2\epsilon}), \widetilde{\Theta}\Big(\frac{d^{1+2\epsilon}}{n_{\mathrm{u}}}\Big)\right\}\cdot\big(\Gamma^{(t)}\big)^{q-1}, i \in [n_{\mathrm{l}}],$$

and

$$\Gamma^{(t+1)} \le (1-\eta\lambda)\cdot\Gamma^{(t)} + \eta\cdot\max\left\{\widetilde{\Theta}(d^{\frac{1}{2}+2\epsilon}), \widetilde{\Theta}\Big(\frac{d^{1+2\epsilon}}{n_{\mathrm{u}}}\Big)\right\}\cdot\big(\Gamma^{(t)}\big)^{q-1},$$

where $\epsilon < 1/8$.

$$\begin{aligned}
\Gamma_l^{(t+1)} &= \langle \mathbf{w}_{j^\star}^{(t+1)}, \boldsymbol{\xi}_l\rangle \\
&= (1-\eta\lambda)\cdot\langle \mathbf{w}_{j^\star}^{(t)}, \boldsymbol{\xi}_l\rangle + \frac{q\eta u_{j^\star}}{n_{\mathrm{l}}+n_{\mathrm{u}}}\left(\sum_{i=1}^{n_{\mathrm{u}}}\widehat{y}_i c_i^{(t)}[\langle \mathbf{w}_{j^\star}^{(t)}, \boldsymbol{\xi}_i\rangle]_+^{q-1}\langle \boldsymbol{\xi}_i, \boldsymbol{\xi}_l\rangle + \sum_{i=1}^{n_{\mathrm{l}}} y_i' b_i^{(t)}[\langle \mathbf{w}_{j^\star}^{(t)}, \boldsymbol{\xi}_i'\rangle]_+^{q-1}\langle \boldsymbol{\xi}_i', \boldsymbol{\xi}_l\rangle\right) \\
&\le (1-\eta\lambda)\cdot\langle \mathbf{w}_{j^\star}^{(t)}, \boldsymbol{\xi}_l\rangle + \frac{q\eta}{n_{\mathrm{l}}+n_{\mathrm{u}}}\Bigg(\underbrace{\sum_{i=1}^{n_{\mathrm{u}}} c_i^{(t)}[\langle \mathbf{w}_{j^\star}^{(t)}, \boldsymbol{\xi}_i\rangle]_+^{q-1}|\langle \boldsymbol{\xi}_i, \boldsymbol{\xi}_l\rangle|}_{\clubsuit} + \underbrace{\sum_{i=1}^{n_{\mathrm{l}}} b_i^{(t)}[\langle \mathbf{w}_{j^\star}^{(t)}, \boldsymbol{\xi}_i'\rangle]_+^{q-1}|\langle \boldsymbol{\xi}_i', \boldsymbol{\xi}_l\rangle|}_{\bigstar}\Bigg),
\end{aligned}$$

Bound ♣ and ★

For ♣, note that $l \in [n_{\mathrm{u}}]$ and there exists an $i \in [n_{\mathrm{u}}]$ equivalent to l, it follows that

$$\begin{aligned}
&\underbrace{\sum_{i=1}^{n_{\mathrm{u}}} c_i^{(t)} [\langle \mathbf{w}_{j^\star}^{(t)}, \boldsymbol{\xi}_i \rangle]_+^{q-1} |\langle \boldsymbol{\xi}_i, \boldsymbol{\xi}_l \rangle|}_{\clubsuit} \\
&= \sum_{i \in [n_{\mathrm{u}}], i \neq l} c_i^{(t)} [\langle \mathbf{w}_{j^\star}^{(t)}, \boldsymbol{\xi}_i \rangle]_+^{q-1} |\langle \boldsymbol{\xi}_i, \boldsymbol{\xi}_l \rangle| + c_l^{(t)} [\langle \mathbf{w}_{j^\star}^{(t)}, \boldsymbol{\xi}_l \rangle]_+^{q-1} \|\boldsymbol{\xi}_l\|_2^2 \qquad \text{(A.20)} \\
&\leq (n_{\mathrm{u}} - 1) \cdot \left(\frac{1}{2} + o(1)\right) \cdot \widetilde{\Theta}(d^{\frac{1}{2}+2\epsilon}) \cdot \left(\Gamma^{(t)}\right)^{q-1} + \left(\frac{1}{2} + o(1)\right) \cdot \widetilde{\Theta}(d^{1+2\epsilon}) \cdot \left(\Gamma^{(t)}\right)^{q-1} \\
&= (n_{\mathrm{u}} - 1) \cdot \widetilde{\Theta}(d^{\frac{1}{2}+2\epsilon}) \cdot \left(\Gamma^{(t)}\right)^{q-1} + \widetilde{\Theta}(d^{1+2\epsilon}) \cdot \left(\Gamma^{(t)}\right)^{q-1},
\end{aligned}$$

For ★, we have

$$\underbrace{\sum_{i=1}^{n_{\mathrm{l}}} b_i^{(t)} [\langle \mathbf{w}_{j^\star}^{(t)}, \boldsymbol{\xi}'_i \rangle]_+^{q-1} |\langle \boldsymbol{\xi}'_i, \boldsymbol{\xi}_l \rangle|}_{\bigstar} \leq n_{\mathrm{l}} \cdot \left(\frac{1}{2} + o(1)\right) \cdot \widetilde{\Theta}(d^{\frac{1}{2}+2\epsilon}) \cdot \left(\Gamma^{(t)}\right)^{q-1} = n_{\mathrm{l}} \cdot \widetilde{\Theta}(d^{\frac{1}{2}+2\epsilon}) \cdot \left(\Gamma^{(t)}\right)^{q-1},$$

Proof Sketch

Lemma 5.1. Assume we use both unlabeled data with pseudo-labels generated by the pseudo-labeler and labeled data for the training of our CNN model. Then for $r \in \{\pm 1\}$, let T_r be the first iteration that $\widehat{\Lambda}_r^{(t)}$ reaches $\Theta(1/m)$, then for $t \in [0, T_r]$, we have

$$\begin{aligned}
\widehat{\Lambda}_r^{(t+1)} &\geq (1-\eta\lambda)\cdot\widehat{\Lambda}_r^{(t)} + \eta\cdot C\cdot\Theta(d)\cdot(\widehat{\Lambda}_r^{(t)})^{q-1}, r \in \{\pm 1\},\\
\bar{\Lambda}_r^{(t+1)} &\leq (1-\eta\lambda)\cdot\bar{\Lambda}_r^{(t)}, r \in \{\pm 1\},\\
\Gamma^{(t+1)} &\leq (1-\eta\lambda)\cdot\Gamma^{(t)} + \eta\cdot\widetilde{\Theta}(d^{1-2\epsilon})\cdot(\Gamma^{(t)})^{q-1},
\end{aligned}$$

Lemma 5.2. Assume we use only labeled data for the training of our CNN model. Then for $i \in [n_l]$, let T_i' be the first iteration that $\Gamma_i'^{(t)}$ reaches $\Theta(1/m)$, then we have

$$\begin{aligned}
\widehat{\Lambda}_r^{(t+1)} &\leq (1-\eta\lambda)\cdot\widehat{\Lambda}_r^{(t)} + \eta\cdot\Theta(d)\cdot\left((\widehat{\Lambda}_r^{(t)})^{q-1} + (\bar{\Lambda}_r^{(t)})^{q-1}\right), r \in \{\pm 1\},\\
\bar{\Lambda}_r^{(t+1)} &\leq (1-\eta\lambda)\cdot\bar{\Lambda}_r^{(t)}, r \in \{\pm 1\},\\
\Gamma_i'^{(t+1)} &\geq (1-\eta\lambda)\cdot\Gamma_i'^{(t)} + \eta\cdot\widetilde{\Theta}(d^{1+2\epsilon})\cdot(\Gamma_i'^{(t)})^{q-1}, i \in [n_l], \text{ for } t \in [0, T_i'].
\end{aligned}$$

Proof Sketch

Lemma 5.3. If both pseudo-labeled and labeled data are used to train CNN, for $r \in \{\pm 1\}$, let T_r be the first iteration that $\widehat{\Lambda}_r^{(t)}$ reaches $\Theta(1/m)$ respectively. Let $T_0 = \max_{r\in\{\pm 1\}}\{T_r\}$. Then, it holds that $\widehat{\Lambda}_r^{(T_0)} = \widetilde{\Theta}(1)$, $\bar{\Lambda}_r^{(t)} = \widetilde{O}(d^{-\frac{1}{4}})$ and $\Gamma^{(t)} = \widetilde{O}(d^{-\frac{1}{4}+\epsilon})$ for all $t \in [0, T_0]$.

Lemma 5.4. If only labeled data are used to train CNN, for $i \in [n_1]$, let T_i' be the first iteration that $\Gamma_i'^{(t)}$ reaches $\Theta(1/m)$. Let $T_0' = \max_{i\in[n_1]} T_i'$. Then, it holds that $\widehat{\Lambda}_r = \widetilde{O}(d^{-\frac{1}{4}})$, $\bar{\Lambda}_r = \widetilde{O}(d^{-\frac{1}{4}})$ for $r \in \{\pm 1\}$ and $\Gamma_i'^{(t)} = \widetilde{\Theta}(1)$ for $i \in [n_1]$.

Lemma 5.5. For any learning rate $\eta = \Theta(1)$, we have $\left\|\mathbf{a}^{(t)}\right\|_1 = \log(t)/\widetilde{\Theta}(1)$. For any labeled data $(\mathbf{x}_i', y_i') \in S'$, we have with high probability that $y_i' \cdot f_{\mathbf{W}^{(t)}}(\mathbf{x}_i') = \left\|\mathbf{a}^{(t)}\right\|_1 \cdot \widetilde{\Theta}(1)$. For any newly generated data $(\mathbf{x}, y) \sim \mathcal{D}$, we also have with high probability that $y \cdot f_{\mathbf{W}^{(t)}}(\mathbf{x}) = \left\|\mathbf{a}^{(t)}\right\|_1 \cdot \widetilde{\Theta}(1)$.

With the help of the above lemma and note that training error and test error are related to $y \cdot f_{\mathbf{W}^{(T_0)}}(\mathbf{x})$ and test loss is related to $\|\mathbf{a}^{(T_0)}\|_1$, we can prove that after $T = \Theta(d^{0.1}/\eta)$ iterations with learning rate $\eta = \Theta(1)$, the model can achieve nearly zero training error, test error, training loss and test loss.

Lemma C.4. Consider an increasing sequence $x_t \geq 0$ defined as $x_{t+1} = x_t + \eta \cdot C_t x_t^{q-1}$, and $C_1 \leq C_t \leq C_2$ for all $t > 0$, then we have for $A > x_0$, every $\delta > 0$, and every $\eta > 0$:

$$\sum_{t \geq 0, x_t \leq A} \eta \leq \frac{\delta}{(1 - (1+\delta)^{-(q-2)}) x_0 C_1} + \eta \cdot \frac{C_2}{C_1} (1+\delta)^{q-1} \left(1 + \frac{\log(A/x_0)}{\log(1+\delta)} \right),$$

$$\sum_{t \geq 0, x_t \leq A} \eta \geq \frac{\delta \left(1 - (x_0/A)^{q-2}\right)}{(1+\delta)^{q-1} \left(1 - (1+\delta)^{-(q-2)}\right) x_0 C_2} - \eta \cdot (1+\delta)^{-(q-1)} \left(1 + \frac{\log(A/x_0)}{\log(1+\delta)} \right).$$