Stochastic Consensus: Enhancing Semi-Supervised Learning with Consistency of Stochastic Classifiers

Hui Tang[1], Lin Sun[2], and Kui Jia[1]

[1] South China University of Technology, Guangzhou, China
eehuitang@mail.scut.edu.cn, kuijia@scut.edu.cn
[2] Magic Leap, Sunnyvale, CA, USA
lsun@magicleap.com

Abstract. Semi-supervised learning (SSL) has achieved new progress recently with the emerging framework of self-training deep networks, where the criteria for selection of unlabeled samples with pseudo labels play a key role in the empirical success. In this work, we propose such a new criterion based on consistency among multiple, stochastic classifiers, termed Stochastic Consensus (STOCO). Specifically, we model parameters of the classifiers as a Gaussian distribution whose mean and standard deviation are jointly optimized during training. Due to the scarcity of labels in SSL, modeling classifiers as a distribution itself provides additional regularization that mitigates overfitting to the labeled samples. We technically generate pseudo labels using a simple but flexible framework of deep discriminative clustering, which benefits from the overall structure of data distribution. We also provide theoretical analysis of our criterion by connecting with the theory of learning from noisy data. Our proposed criterion can be readily applied to self-training based SSL frameworks. By choosing the representative FixMatch as the baseline, our method with multiple stochastic classifiers achieves the state of the art on popular SSL benchmarks, especially in label-scarce cases.

Keywords: Semi-supervised learning, stochastic classifiers, consistency criterion, deep discriminative clustering

1 Introduction

The practical success of deep learning across a range of application problems assumes the access to massive amounts of annotated training data. However, data annotations are usually costly, and in some cases they could be difficult to be acquired due to, e.g., the lack of domain expertise. The situation motivates topics of data-efficient learning, such as semi-supervised learning (SSL) [25], few-shot learning [15], and domain adaptation [57], etc. Among them, SSL is a more classical one that aims for model learning with a few number of labeled samples and a large number of unlabeled ones from the same data distribution.

Deep SSL achieves good progress recently, and the methods generally fall in three categories. Those in the first category are based on self-training with

pseudo labels [18,25]; they work better by selecting unlabeled samples with pseudo labels assigned by the previously learned model, and then updating the model in a supervised manner using both the labeled and pseudo-labeled data; the selection criteria are usually based on confidence filtering of pseudo labels [16,26,44,53], where the unlabeled samples with high confidence remain and others are discarded. Methods in the second category are based on consistency regularization that enforces the consistency of model predictions between a sample and its perturbed counterpart, including randomly augmented duplicates [42], virtual adversarial examples [33], similar/smooth neighbors [30], to name a few; the smoothness assumption is also considered in these methods, i.e., close input samples should have close labels. The last category of FixMatch [44] and CoMatch [26] has shown remarkable performance by integrating self-training and consistency regularization. In spite of these advances, we show in this paper that the selection criteria in existing methods can be further improved for better SSL.

Specifically, we propose a novel consistency criterion among multiple stochastic classifiers under the self-training framework of SSL, termed Stochastic Consensus (STOCO); Fig. 1 gives the illustration. The proposed criterion is partially inspired by co-training [3,6,8] and tri-training [41,55]; they leverage category predictions of one or two classifiers on unlabeled samples to enlarge the training set, wherein a design principle is based on majority voting that shares a similar insight with the popular techniques of ensemble learning [7,12]. In classical ensemble learning, the number of model parameters grows linearly with that of model classifiers. To improve the efficiency, we propose the use of stochastic classifiers [29] for consistency criterion, where parameters of multiple stochastic classifiers are sampled from a same Gaussian distribution whose mean and standard deviation are simultaneously optimized during training. In the extreme case, one can sample an infinite number of classifiers while keeping the model size unchanged. Due to the scarcity of labels in SSL, modeling classifiers as a distribution itself can provide regularization that mitigates overfitting to the labeled samples. We note that a recent work UPS [39] uses MC-dropout [17] to model randomness, which is model-dependent and data-independent, and yields diverse network structures; differently, STOCO uses stochastic classifiers, whose parameters are modeled as a learnable, model-agnostic Gaussian distribution that can *dynamically capture the pattern of sensible decision boundaries*, directly benefiting model generalization, as demonstrated in Sec. 4.

To implement our proposed consistency criterion, for any unlabeled sample, we compute the element-wise product of category predictions from multiple stochastic classifiers, and select samples with the maximum value in the product higher than a pre-defined threshold; we then take an average of the predictions from multiple classifiers, and generate pseudo labels from the thus obtained averages using a simple but flexible deep learning based discriminative clustering framework [13]. Intuitively, the pseudo labels are generated to both encourage the cluster size balance and respect the underlying data distribution. In this work, we provide theoretical analysis of our proposed criterion by connecting with the theory of learning from noisy data [1]. Our method can be readily

applied to self-training based SSL frameworks. Choosing the representative FixMatch as the baseline, our method with multiple stochastic classifiers achieves the state of the art on four popular benchmarks, especially in label-scarce cases.

2 Related Works

Over the past two decades, a huge literature has emerged on semi-supervised learning (SSL), including a broad variety of algorithms [14]. We focus on deep learning based ones, which can be mainly divided into three categories except for those meta-learning based SSL approaches [19,27,36,47].

Methods in the first category leverage the idea from the earlier works [32,43] that pseudo labels for unlabeled data are produced by the trained model itself and then used to refine the current model, termed as self-training. Such a simple strategy is widely adopted or developed in various fields [16,18,25,39,40,51,56,57]. Lee [25] assigns the highest-score category to an unlabeled sample. Entropy minimization [18] directly uses predicted class probability distributions as pseudo labels. To improve, confidence thresholding [16] is often used to select reliable pseudo labels. Recently, UPS [39] utilizes both uncertainty and confidence of a network prediction to select a more accurate subset of pseudo labels.

The second category is consistency regularization, which enforces the consistent prediction between a sample and its counterpart perturbed by modifying the model [24,38,46,54] or input [2,30,33,42]. For example, Rasmus et al. [38] minimize the difference between the activations of the unperturbed parent model and those of the perturbed models after denoising; Mean Teacher [46] is the moving average over weights of model parameters, whose predictions are used as targets; WCP [54] derives the worst-case perturbations on network weights and structures via optimizing with spectral methods and then stabilizes model predictions in presence of such perturbations; Sajjadi et al. [42] make the model prediction consistent for an individual unlabeled sample when it goes through multiple passes of random transformation; VAT [33] reduces the divergence between model predictions of the vanilla unlabeled sample and its virtual adversarial counterpart; SNTG [30] constructs a graph based on mean teacher predictions to guide the student model so that the neighbors have similar features; PAWS [2] enforces the consistency of distance-based, non-parametrically predictions between the anchor and positive views of a same unlabeled image.

The last category includes methods combining self-training with consistency regularization [4,5,26,44,50,53]. Apart from mixup, ReMixMatch [4] encourages alignment between marginal class distributions of labeled and unlabeled data, and makes consistent predictions between a weakly-augmented image and multiple strongly-augmented images of the same sample. FixMatch [44] trains the model on the strongly-augmented version of a sample with the prediction on its weakly-augmented version as the pseudo label; pseudo labels are selected above a pre-defined confidence threshold. Flexibly adjusting class-wise confidence thresholds is introduced in [53], where the principle is to scale down the fixed threshold if one class has less highly confident samples. CoMatch [26] uti-

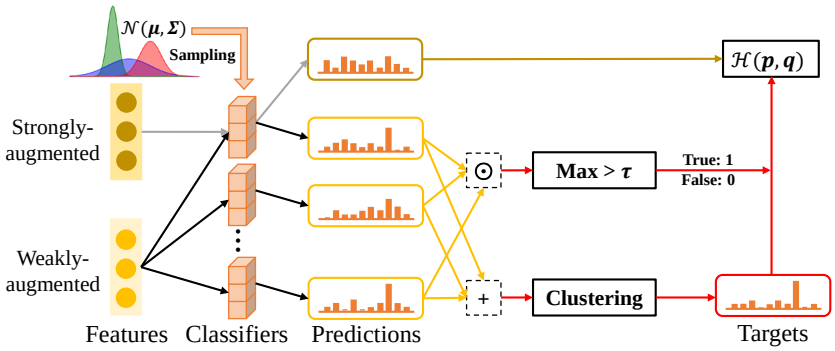


Fig. 1: Diagram for our method of stochastic consensus (STOCO). We sample multiple classifiers from a learned Gaussian distribution $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$; for the weakly-augmented version of any unlabeled sample, we calculate the element-wise product of category predictions from these stochastic classifiers and select samples with the maximum value in the product higher than a pre-defined threshold τ; we take an average over the predictions from multiple classifiers, and generate pseudo labels from the thus obtained averages via deep discriminative clustering; then, with these derived targets, the model is trained using the strongly-augmented version of selected samples via a cross-entropy loss $\mathcal{H}(\boldsymbol{p}, \boldsymbol{q})$.

lizes the similarities between embeddings of unlabeled samples to weight and sum their class probabilities as the pseudo label; a pseudo label graph is then constructed to regularize the embedding graph. Our method shares a similar motivation with these ones, but differs in the aim to progressively improve the noise rate of selected samples by applying the proposed consistency criterion among multiple stochastic classifiers, in a distinctive perspective of designing a more strict criterion, which is under-explored.

3 The Proposed Method

Consider a labeled batch with n_x pairs of samples and one-hot labels $\mathcal{X} = \{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^{n_x}$, and an unlabeled batch with n_u samples $\mathcal{U} = \{\boldsymbol{u}_i\}_{i=1}^{n_u}$, where $n_u = \gamma n_x$. Here, γ controls the relative size of $\mathcal{X}$ and $\mathcal{U}$. Let the number of classes be K. The objective of semi-supervised learning (SSL) is to predict class labels for unseen samples by learning a feature extractor $g(\cdot; \boldsymbol{\theta}_g)$ that lifts any input sample to the feature space $\mathcal{Z}$, and a classifier $f(\cdot; \boldsymbol{\theta}_f)$ that outputs class probabilities from the feature $\boldsymbol{z} \in \mathcal{Z}$, where $\boldsymbol{\theta}_g$ and $\boldsymbol{\theta}_f$ collect the network parameters of feature extractor and classifier respectively. Let $p(\mathbf{x}; \boldsymbol{\theta}_m)$ be the label distribution predicted by the classification model $f(g(\cdot))$, where $\boldsymbol{\theta}_m = \{\boldsymbol{\theta}_g, \boldsymbol{\theta}_f\}$ collects all parameters of the model. Let $\mathcal{H}(\cdot, \cdot)$ be the cross entropy between two probability distributions. For unlabeled data, we consider two types of data augmentation: a strong one (i.e., RandAugment [11]) and a weak one (i.e., standard flip-and-shift strategy), denoted by $\mathcal{A}(\cdot)$ and $\alpha(\cdot)$ respectively.

3.1 FixMatch

FixMatch [44] integrates two simple but effective SSL techniques, self-training [25] and consistency regularization [42]. The recent theoretical result [48] has suggested that such a combination could achieve high accuracy concerning ground-truth labels. Specifically, FixMatch optimizes two losses: a supervised loss $\mathcal{L}_s$ and an unsupervised loss $\mathcal{L}_u$, computed on $\mathcal{X}$ and $\mathcal{U}$ respectively. $\mathcal{L}_s$ is the cross entropy between predicted label distribution and ground-truth one, computed on the weakly-augmented labeled images:

$$\mathcal{L}_s = \frac{1}{n_x}\sum_{i=1}^{n_x}\mathcal{H}(p(\alpha(\boldsymbol{x}_i);\boldsymbol{\theta}_m),\boldsymbol{y}_i). \tag{1}$$

The computation of $\mathcal{L}_u$ is as follows. For the weakly-augmented unlabeled images $\{\alpha(\boldsymbol{u}_i)\}_{i=1}^{n_u}$, FixMatch first produces category predictions $\{p(\alpha(\boldsymbol{u}_i);\boldsymbol{\theta}_m)\}_{i=1}^{n_u}$; the images with $\max p(\alpha(\boldsymbol{u}_i);\boldsymbol{\theta}_m) > \tau$ are selected and their pseudo labels are generated by $\hat{y}_i = \arg\max p(\alpha(\boldsymbol{u}_i);\boldsymbol{\theta}_m)$, where the hyperparameter τ determines the threshold of traditional confidence filtering [16,40,44,50,57]; then, the model is trained on the strongly-augmented version of the selected samples to predict the generated pseudo labels. For any selected sample $\boldsymbol{u}_i$, we denote the one-hot pseudo label of $\alpha(\boldsymbol{u}_i)$ as $\hat{\boldsymbol{y}}_i$ and write $\mathcal{L}_u$ as the cross entropy between predicted label distribution of $\mathcal{A}(\boldsymbol{u}_i)$ and $\hat{\boldsymbol{y}}_i$:

$$\mathcal{L}_u=\frac{1}{n_u}\sum_{i=1}^{n_u}\mathrm{I}[\max p(\alpha(\boldsymbol{u}_i);\boldsymbol{\theta}_m)>\tau]\mathcal{H}(p(\mathcal{A}(\boldsymbol{u}_i);\boldsymbol{\theta}_m),\hat{\boldsymbol{y}}_i), \tag{2}$$

where $\mathrm{I}[\cdot]$ is an indicator. Combining $\mathcal{L}_s$ and $\mathcal{L}_u$ gives the overall loss of FixMatch:

$$\mathcal{L}_{overall} = \mathcal{L}_s + \lambda_u\mathcal{L}_u, \tag{3}$$

where λ_u is to make a trade-off in the joint optimization problem. Optimizing Eq. (3) implements self-training and consistency regularization simultaneously.

3.2 Our Method: Stochastic Consensus

FixMatch achieves the state-of-the-art performance by setting a high τ, which improves the quality of pseudo labels. However, its use of the traditional confidence criterion is sub-optimal since it only leverages the prediction information from one classifier. The earlier works of co-training [8] and tri-training [55] have suggested that the prediction information from other classifiers can be helpful. To this end, we propose for SSL a novel sample selection scheme, the consistency criterion among multiple classifiers, which can further improve the quality of pseudo labels. Generally, the model size will linearly increase as the number of classifiers grows. To enhance efficiency, we propose to use stochastic classifiers [29], which are modeled by a Gaussian distribution whose parameters are jointly learned in training. One can sample an arbitrary number of classifiers from the learned distribution while keeping the model size consistent. We thus term our

proposed method as Stochastic Consensus (STOCO). Moreover, we use a simple but flexible deep learning based discriminative clustering framework [13] to generate a soft version of pseudo labels, which encourages cluster size balance while avoiding the introduction of additional hyperparameter: temperature T [4,5].

Consistency Criterion among Stochastic Classifiers. A Gaussian distribution $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ is used to model the classifier $f(\cdot; \boldsymbol{\theta}_f)$, i.e., $\boldsymbol{\theta}_f \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, where $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ are the mean vector (by flattening the weight matrix) and diagonal covariance matrix respectively. With the reparametrisation trick [22], the overall loss will be back-propagated to the learnable parameters of $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$. With diagonal $\boldsymbol{\Sigma}$, we can always keep the model size as having two classifiers. To be specific, we first draw m vectors $\{\boldsymbol{\epsilon}_j\}_{j=1}^m$ of the same size as $\boldsymbol{\mu}$ from a standard Gaussian distribution. Then, m stochastic classifiers $\{\boldsymbol{\theta}_f^j\}_{j=1}^m$ are derived by:

$$\boldsymbol{\theta}_f^j = \boldsymbol{\mu} + \boldsymbol{\sigma} \odot \boldsymbol{\epsilon}_j, \tag{4}$$

where $\boldsymbol{\sigma}$ is the diagonal of $\boldsymbol{\Sigma}$ and $\odot$ indicates element-wise product. Due to the nature of randomness in the independent sampling, classifiers in $\{\boldsymbol{\theta}_f^j\}_{j=1}^m$ are different; meanwhile, they do not deviate too much from each other since they come from the same source, thus stabilizing training and avoiding degeneration. On the other hand, since SSL only has access to a few labeled samples, a set of sensible solutions might exist and thus it is natural to model the classifier as a distribution, which also acts as an implicit regularization to mitigate overfitting. These characteristics provide sufficient conditions for our consistency criterion. For a weakly-augmented version of a given unlabeled sample $\boldsymbol{u}_i$, we extract the feature by $\boldsymbol{z}_i^\alpha = g(\alpha(\boldsymbol{u}_i); \boldsymbol{\theta}_g)$. Each classifier in $\{\boldsymbol{\theta}_f^j\}_{j=1}^m$ takes $\boldsymbol{z}_i^\alpha$ as input and outputs the class probability distribution $p(\boldsymbol{z}_i^\alpha; \boldsymbol{\theta}_f^j)$. We take an ensemble of the predicted label distributions by computing the element-wise product:

$$\dot{p}(\boldsymbol{z}_i^\alpha) = p(\boldsymbol{z}_i^\alpha; \boldsymbol{\theta}_f^1) \odot p(\boldsymbol{z}_i^\alpha; \boldsymbol{\theta}_f^2) \odot \cdots p(\boldsymbol{z}_i^\alpha; \boldsymbol{\theta}_f^m). \tag{5}$$

The sample will be selected if $\max \dot{p}(\boldsymbol{z}_i^\alpha) > \tau$. The proposed criterion is in fact an evolved version of self-training, which selects unlabeled samples for one classifier if all classifiers are confident of the same class. Such a selection strategy is essentially majority voting, leading to more reliable results [7,12,37]. Without loss of generality, the cross entropy in $\mathcal{L}_u$ is computed between $p(\boldsymbol{z}_i^\mathcal{A}; \boldsymbol{\theta}_f^1)$ and the one-hot form of $\hat{y}_i = \arg\max \dot{p}(\boldsymbol{z}_i^\alpha)$, where $\boldsymbol{z}_i^\mathcal{A} = g(\mathcal{A}(\boldsymbol{u}_i); \boldsymbol{\theta}_g)$.

Pseudo Label Generation via Deep Discriminative Clustering. The core idea is to introduce an auxiliary distribution [13,20,49] by considering the overall data structure in the feature space, which enforces *structural regularization*. Specifically, given m predicted label distributions $\{p(\boldsymbol{z}_i^\alpha; \boldsymbol{\theta}_f^j)\}_{j=1}^m$ for the weakly-augmented version of an unlabeled sample $\boldsymbol{u}_i$, we take the average as:

$$\bar{p}(\boldsymbol{z}_i^\alpha) = \frac{1}{m} \sum_{j=1}^{m} p(\boldsymbol{z}_i^\alpha; \boldsymbol{\theta}_f^j), \tag{6}$$

which integrates the prediction information from all m stochastic classifiers and is still a probability distribution (i.e., $\sum \bar{p}(\boldsymbol{z}_i^\alpha) = 1$). For unlabeled data $\{\boldsymbol{u}_i\}_{i=1}^{n_u}$,

we collectively write the predicted probability vectors $\{\bar{\boldsymbol{p}}_i^\alpha\}_{i=1}^{n_u}$ as $\boldsymbol{P}$, where $\bar{\boldsymbol{p}}_i^\alpha = \bar{p}(\boldsymbol{z}_i^\alpha)$. We also write $\bar{p}_{ik}^\alpha$ as the k-th element of $\bar{\boldsymbol{p}}_i^\alpha$. To refine the model predictions iteratively, we then introduce a target counterpart $\boldsymbol{Q} = \{\bar{\boldsymbol{q}}_i^\alpha\}_{i=1}^{n_u}$, which is obtained by optimizing the following objective [13]:

$$\min_{\boldsymbol{Q}} \mathrm{KL}(\boldsymbol{P}|\boldsymbol{Q}) + \mathrm{KL}(\boldsymbol{\varrho}|\boldsymbol{\pi}), \tag{7}$$

where $\boldsymbol{\varrho} = 1/n_u \sum_{i=1}^{n_u} \bar{\boldsymbol{q}}_i^\alpha$ is the empirical label distribution, $\boldsymbol{\pi}$ is the uniform distribution, and $\mathrm{KL}(\cdot|\cdot)$ denotes the KL divergence between two distributions. The first term in Eq. (7) minimizes the divergence between $\boldsymbol{P}$ and $\boldsymbol{Q}$, which avoids the targets deviating too much from the predictions and thus shows respect to the underlying data distribution; the second term minimizes the divergence between $\boldsymbol{\varrho}$ and $\boldsymbol{\pi}$, which avoids degenerate solutions (i.e., cluster merging) and encourages cluster size balance. The closed-form solution of $\boldsymbol{Q}$ is derived by [13]:

$$\bar{q}_{ik}^\alpha = \frac{\bar{p}_{ik}^\alpha / (\sum_{i'} \bar{p}_{i'k}^\alpha)^{0.5}}{\sum_{k'} \bar{p}_{ik'}^\alpha / (\sum_{i'} \bar{p}_{i'k'}^\alpha)^{0.5}}, \tag{8}$$

which generates the pseudo labels to supervise the model learning, whose effectiveness has been demonstrated in various applications [9,21,28,34,45].

Given $\dot{p}(\boldsymbol{z}_i^\alpha)$ and $\bar{\boldsymbol{q}}_i^\alpha$ for any unlabeled sample, we have an improved version of the unsupervised loss $\mathcal{L}_u$ as:

$$\mathcal{L}_u = \frac{1}{n_u} \sum_{i=1}^{n_u} \mathrm{I}[\max \dot{p}(\boldsymbol{z}_i^\alpha) > \tau] \mathcal{H}(p(\boldsymbol{z}_i^{\mathcal{A}}; \boldsymbol{\theta}_f^1), \bar{\boldsymbol{q}}_i^\alpha). \tag{9}$$

3.3 Theoretical Analysis

We provide theoretical analysis for our method to show its progressively improved classification error by connecting with the theory in [1]. The work [1] adapts the probably approximately correct (PAC) learning theory from reliable data to noisy data, giving a learning algorithm guidance on how to handle incorrect training samples. The theory is explained below.

Theorem 1. *([1], Theorem 2) If we draw a sequence ς of*

$$\zeta \geq \frac{2}{\epsilon^2 (1-2\eta)^2} \ln(\frac{2N}{\delta}) \tag{10}$$

samples, then a hypothesis h that minimizes the disagreement with ς will have the PAC property:

$$\Pr[d(h, h^*) \geq \epsilon] \leq \delta, \tag{11}$$

where ϵ is the classification error rate of the worst-case hypothesis, η (< 0.5) is an upper bound on the classification noise rate, N is the number of hypotheses, δ is a confidence parameter, and $d(\cdot, \cdot)$ is the sum over the probability of elements from the symmetric difference between hypotheses h and h^* (the ground truth).

Theorem 1 tells that if the condition (10) is satisfied, then the difference between the conjectured rule h and the correct rule h^* will be small (less than ϵ) with a high probability (greater than $1-\delta$). Following [55], we have:

$$v = \frac{c}{\epsilon^2} = \zeta(1-2\eta)^2, \tag{12}$$

where $c = 2\nu \ln(\frac{2N}{\delta})$, v is an intermediate variable, and ν is a positive number to make the condition (10) hold equality. During the model training of our method, the classification noise rate at the t-th iteration is estimated by:

$$\eta^t = \frac{\eta_x^t|\mathcal{X}| + \check{\eta}_u^t|\mathcal{U}_l^t|}{|\mathcal{X} \cup \mathcal{U}_l^t|}, \tag{13}$$

where η_x^t denotes the classification noise rate on the labeled set $\mathcal{X}$, $\eta_x^t|\mathcal{X}|$ is accordingly the number of labeled samples mislabeled by the model, $\mathcal{U}_l^t$ indicates the set of unlabeled samples selected by our method, $\check{\eta}_u^t$ denotes the estimation of the upper-bound classification noise rate on $\mathcal{U}_l^t$, and $\check{\eta}_u^t|\mathcal{U}_l^t|$ is accordingly the number of mislabeled samples in $\mathcal{U}_l^t$. According to Eq. (12), v^t is computed by:

$$v^t = \zeta^t(1-2\eta^t)^2 = |\mathcal{X} \cup \mathcal{U}_l^t| \left(1 - 2\frac{\eta_x^t|\mathcal{X}| + \check{\eta}_u^t|\mathcal{U}_l^t|}{|\mathcal{X} \cup \mathcal{U}_l^t|}\right)^2. \tag{14}$$

Eq. (12) shows that v is proportional to $1/\epsilon^2$, i.e., $\epsilon^t < \epsilon^{t-1}$ if $v^t > v^{t-1}$, suggesting that the classification model can be progressively improved via the use of $\mathcal{U}_l^t$ in training. The condition $v^t > v^{t-1}$ can be also written as:

$$|\mathcal{X} \cup \mathcal{U}_l^t| \left(1 - 2\frac{\eta_x^t|\mathcal{X}| + \check{\eta}_u^t|\mathcal{U}_l^t|}{|\mathcal{X} \cup \mathcal{U}_l^t|}\right)^2 > |\mathcal{X} \cup \mathcal{U}_l^{t-1}| \left(1 - 2\frac{\eta_x^{t-1}|\mathcal{X}| + \check{\eta}_u^{t-1}|\mathcal{U}_l^{t-1}|}{|\mathcal{X} \cup \mathcal{U}_l^{t-1}|}\right)^2. \tag{15}$$

Considering that η_x^t and η_x^{t-1} can be very small and assuming that $0 < \check{\eta}_u^t, \check{\eta}_u^{t-1} < 0.5$, we can simplify the condition (15) as:

$$0 < \frac{\check{\eta}_u^t}{\check{\eta}_u^{t-1}} < \frac{|\mathcal{U}_l^{t-1}|}{|\mathcal{U}_l^t|} < 1. \tag{16}$$

Our consistency criterion among multiple stochastic classifiers conducts a strict selection process, where one unlabeled sample will be selected if all classifiers have consistent and confident predictions, leading to a lower classification noise rate than the traditional confidence filter used in FixMatch. It implies that the assumption of $0 < \check{\eta}_u^t, \check{\eta}_u^{t-1} < 0.5$ would be implemented and thus the condition of $\check{\eta}_u^t < \check{\eta}_u^{t-1}$ would be met. On the other hand, the number of unlabeled samples selected by our method would increase in a gradual manner with the training due to the strict selection; consequently, the conditions of $|\mathcal{U}_l^t| > |\mathcal{U}_l^{t-1}|$ and $\check{\eta}_u^t|\mathcal{U}_l^t| < \check{\eta}_u^{t-1}|\mathcal{U}_l^{t-1}|$ would be satisfied. These analyses suggest that our method would iteratively improve the model performance, as demonstrated in Sec. 4.2.

4 Experiments

In this section, we follow FixMatch [44] in terms of hyperparameter setting and model architecture, and evaluate our method using FixMatch as the backbone on typical semi-supervised learning (SSL) benchmark datasets.

4.1 Setup

Datasets. We use the following four SSL benchmark datasets for our experiments with various number of labeled samples. ***CIFAR-10*** [23] contains $60,000$ colour images in 10 classes, with $6,000$ images per class. There are $50,000$ training images and $10,000$ test images. The 10 classes are completely mutually exclusive. We follow FixMatch and examine on the three settings of 40, 250, and $4,000$ labels. ***CIFAR-100*** [23] also has $60,000$ images in total but consists of 100 classes, resulting in a more chanllenging classification scenario. Each class has 600 images, where 500 images are for training and the remaining 100 are for testing. We experiment on 400-, $2,500$-, and $10,000$-label settings. ***SVHN*** [35] includes $73,257$ images for training and $26,032$ images for testing. There are 10 classes in SVHN, corresponding to 10 digits of $\{0, 1, \ldots, 9\}$. Images in SVHN have a colored background and multiple extremely blurred digits, which are taken from the real-world streets. We evaluate on the three settings of 40, 250, and $1,000$ labels. ***STL-10*** [10] is a dataset tailored for SSL, which comprises $5,000$ labeled color images of size 96×96 and $100,000$ unlabeled images. The unlabeled samples are drawn from a distribution slightly shifted from the one of labeled data, leading to a more realistic test. The labeled set is split into ten pre-defined folds of $1,000$ images each. We evaluate on five of these ten folds.
Implementation Details. For all experiments, we follow FixMatch's training protocol, including optimizer, learning rate schedule, and data preprocessing, and consistently apply the same hyperparameter setting, e.g., $\gamma = 7$, $\tau = 0.95$, and $\lambda_u = 1$. Besides, we empirically set the classifier number m as 5. For CIFAR-10 and SVHN, we use a Wide ResNet-28-2 [52] as the base network; for CIFAR-100, we use a Wide ResNet-28-8 that leverages more convolution filters to cope with larger label space; for STL-10, we use a Wide ResNet-37-2 that utilizes more convolution layers to handle higher input resolution. For inference, we use a fixed classifier determined by the learned mean $\boldsymbol{\mu}$ and report the classification result of mean±std over five trials with different folds of labeled data.

4.2 Ablation Studies and Learning Analyses

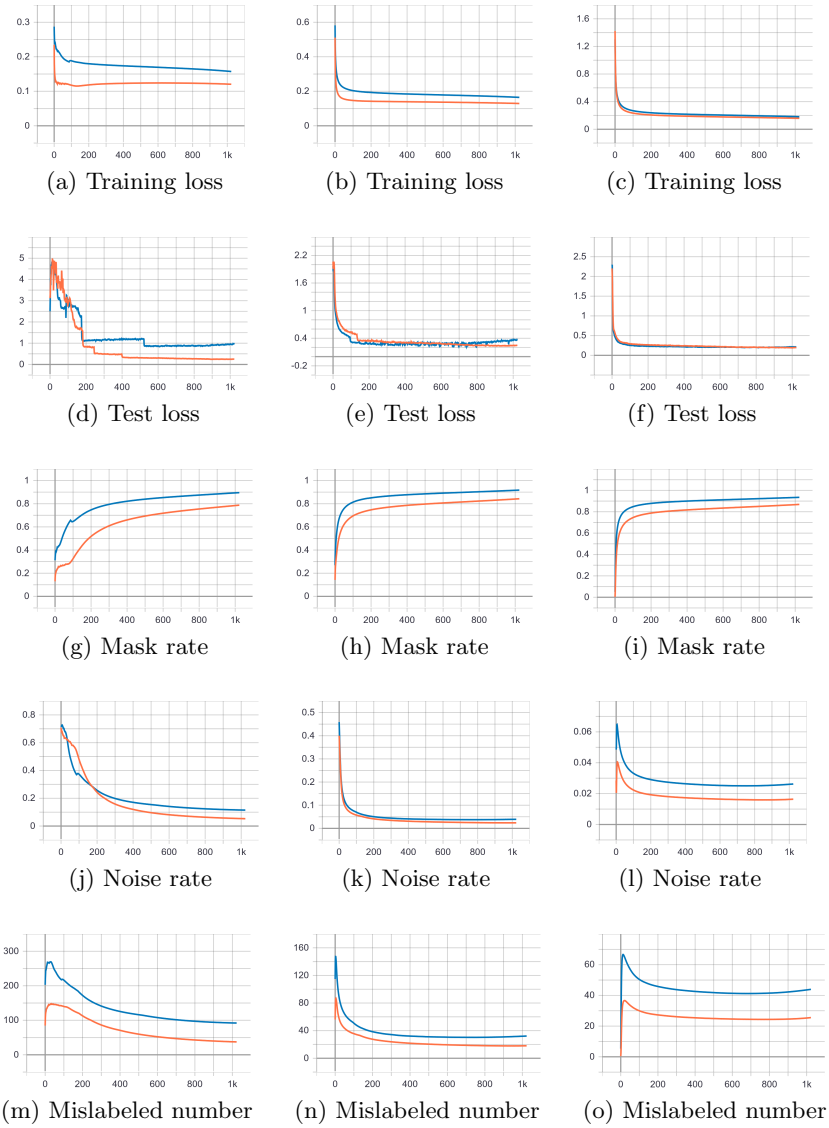
Ablation Studies. To examine the effects of two key components in our method, we conduct careful ablation studies on CIFAR-10 with 40 labels by evaluating several variants of our method: **(1)** STOCO (w/o CC and DDC), which removes both consistency criterion among stochastic classifiers and pseudo label generation via deep discriminative clustering, namely FixMatch; **(2)** STOCO (w/o CC), which removes the consistency criterion only; **(3)** STOCO with varied number of stochastic classifiers, i.e., $m \in \{1, 2, 5, 10, 15, 20\}$. Results are shown

Method	STOCO (w/o CC and DDC)	STOCO (w/o CC)	STOCO (m=1)	STOCO (m=2)	STOCO (m=5)	STOCO (m=10)	STOCO (m=15)	STOCO (m=20)
Error rate	11.27	9.25	8.46	6.68	**4.74**	4.86	6.79	7.19

Table 1: Ablation studies. We follow [4,26,44] to report error rates on a single 40-label split from CIFAR-10. STOCO (w/o CC and DDC) removes both consistency criterion among stochastic classifiers and pseudo label generation via deep discriminative clustering, namely FixMatch. STOCO (w/o CC) removes the consistency criterion only. STOCO ($m = 5$) is with 5 stochastic classifiers, i.e., our method.

in Table 1. We observe that our method ($m = 5$) degrades by 4.51% after removing the consistency criterion, and then by 2.02% after successively removing the deep discriminative clustering. This verifies that both components are indispensable and thus our method has a reasonable design. Given that the only difference between STOCO ($m = 1$) and STOCO (w/o CC) is whether they use a stochastic classifier, the former slightly outperforms the latter, showing the superiority of the stochastic classifier. The error rate decreases with the growth of m when $m \leq 5$, indicating that more classifiers can enhance the generalization ability via more strict sample selection; a reverse phenomenon is observed when $m > 5$, suggesting that the selection process is too strict to involve enough unlabeled samples in training so that the model cannot converge fast and well. Notably, our STOCO yields fairly stable performance when m varies in a wide range, suggesting the excellent robustness of our method.

Learning Analyses. As analyzed in Sec. 3.3, the proposed consistency criterion among multiple stochastic classifiers conducts a strict selection process, which would meet the three conditions of progressively improving model's generalization ability: **(1)** $|\mathcal{U}_l^t| > |\mathcal{U}_l^{t-1}|$, which states that the number of selected unlabeled samples should increase as the training proceeds; **(2)** $\check{\eta}_u^t < \check{\eta}_u^{t-1}$, which tells that the classification noise rate of the selected pseudo-labeled set should decrease iteratively; **(3)** $\check{\eta}_u^t|\mathcal{U}_l^t| < \check{\eta}_u^{t-1}|\mathcal{U}_l^{t-1}|$, which describes that the number of mislabeled samples in the selected pseudo-labeled set should reduce with the training. To verify these empirically, we conduct experiments on CIFAR-10 with 40, 250, and $4,000$ labels, and examine how the following five quantities evolve during training; they are the training loss measured on the training set, test loss measured on the test set, mask rate that is the ratio of selected samples in an unlabeled batch, noise rate and mislabeled number that are respectively the ratio and number of incorrectly labeled samples in the selected pseudo-labeled set, which are measured using the ground truth labels, just for visualization. In Fig. 2, we plot the evolving curves of these quantities during training, by comparing our method with the baseline FixMatch [44]. We highlight several observations below. **(1)** The training loss of our STOCO is lower than that of FixMatch since our proposed consistency criterion selects fewer unlabeled samples, which are the most confident ones with the largest easiness. **(2)** The test loss of our STOCO decreases and then stabilizes at a low level, showing the *progressively improved generalization performance*; particularly, in the extreme case where only 4 la-



(a) Training loss (b) Training loss (c) Training loss

(d) Test loss (e) Test loss (f) Test loss

(g) Mask rate (h) Mask rate (i) Mask rate

(j) Noise rate (k) Noise rate (l) Noise rate

(m) Mislabeled number (n) Mislabeled number (o) Mislabeled number

Fig. 2: Learning analyses on our STOCO. For all subfigures, the horizontal axis represents the training epoch; the colors of blue and orange correspond to the results of FixMatch and our method respectively. The results are obtained on CIFAR-10 with 40 (column 1), 250 (column 2), and 4,000 (column 3) labels. Refer to the main text for how these quantities are defined and computed.

bels are available per class (cf. Fig. 2d), our STOCO exhibits a clear loss drop with the training when compared to FixMatch, verifying the better convergence

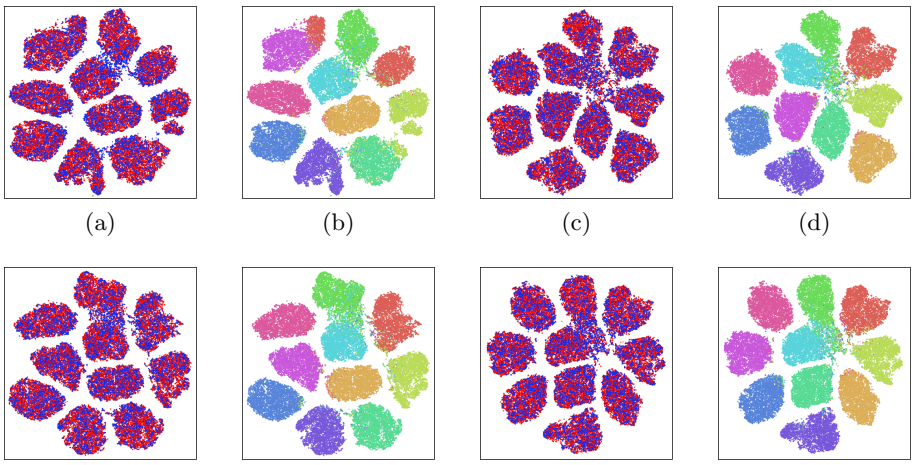


Fig. 3: The t-SNE visualization of features learned by FixMatch (left two columns) and our STOCO (right two columns). In columns 1 and 3, the colors of red and blue denote the training and test samples respectively; their counterparts whose classes are color-coded are in columns 2 and 4 respectively. Results in these plots are obtained on CIFAR-10 with 40 (a-d) and 250 (e-h) labels.

performance of our method. **(3)** In the row of test loss, we find that FixMatch suffers a slight rise at the late stage of training whereas our STOCO does not, suggesting that *our method indeed has the effect of alleviating overfitting*. **(4)** As the training process proceeds, our STOCO has an increasing mask rate, and its noise rate and mislabeled number decrease, indicating that the three conditions are satisfied; notably, *our method achieves a much lower mislabeled number than FixMatch on all label settings*, demonstrating the superiority of our method. These observations corroborate our analyses in Sec. 3.3.

Feature Visualization. To get an intuitive sense of the effect of our method, we expose qualitative differences between the strong baseline FixMatch [44] and our method. We use t-SNE [31] to visualize features of both training and test data, which are extracted by the learned feature extractor of each method. The results on CIFAR-10 with 40 and 250 labels are plotted in Fig. 3. We emphasize several interesting observations below. **(1)** The marginal feature distributions of training and test data are similar, i.e., test samples lie in the support of training data (cf. columns 1 and 3). **(2)** As the size of the labeled set increases, the class-conditional feature distribution becomes purer and gets closer to the true label distribution (cf. columns 2 and 4), suggesting that the model generalization improves. **(3)** On the extremely label-scarce setting (cf. Fig. 3a and Fig. 3c), our STOCO yields more similar marginal feature distributions between the training and test data. **(4)** In Fig. 3b of FixMatch, two different classes wrongly merge into one cluster, e.g., red airplane and purple ship. A possible reason is that the

	CIFAR-10			CIFAR-100			SVHN			STL-10
Method	40 labels	250 labels	4000 labels	400 labels	2500 labels	10000 labels	40 labels	250 labels	1000 labels	1000 labels
Π-Model [38]	-	54.26±3.97	14.01±0.38	-	57.25±0.48	37.88±0.11	-	18.96±1.92	7.54±0.36	26.23±0.82
Pseudo-Labeling [25]	-	49.78±0.43	16.09±0.28	-	57.38±0.46	36.21±0.19	-	20.21±1.09	9.94±0.61	27.99±0.83
Mean Teacher [46]	-	32.32±2.30	9.19±0.19	-	53.91±0.57	35.83±0.24	-	3.57±0.11	3.42±0.07	21.43±2.39
MixMatch [5]	47.54±11.50	11.05±0.86	6.42±0.10	67.61±1.32	39.94±0.37	28.31±0.33	42.55±14.53	3.98±0.23	3.50±0.28	10.41±0.61
UPS [39]	-	-	6.42	-	-	-	-	-	-	-
Meta-Semi [47]	-	-	6.10±0.10	-	-	29.69±0.18	-	-	-	8.03±0.24
UDA [50]	29.05±5.93	8.82±1.08	4.88±0.18	59.28±0.88	33.13±0.22	24.50±0.25	52.63±20.51	5.69±2.76	2.46±0.24	7.66±0.56
ReMixMatch [4]	19.10±9.64	5.44±0.05	4.72±0.13	44.28±2.06	27.43±0.31	23.03±0.56	*3.34*±0.20	2.92±0.48	2.65±0.08	**5.23**±0.45
FixMatch [44]	13.81±3.37	5.07±0.65	4.26±0.05	48.85±1.75	28.29±0.11	22.60±0.12	3.96±2.17	*2.48*±0.38	**2.28**±0.11	7.98±1.50
CoMatch [26]	*6.91*±1.39	4.91±0.33	-	-	-	-	-	-	-	-
FlexMatch [53]	**4.99**±0.16	*4.80*±0.06	*3.95*±0.03	**32.44**±1.99	**23.85**±0.23	**19.92**±0.06	5.36±2.38	-	2.86±0.91	*5.56*±0.22
STOCO	7.17±1.95	**4.77**±0.30	**3.86**±0.05	*41.45*±1.21	*27.41*±0.35	*21.82*±0.20	**2.85**±0.16	**2.47**±0.14	*2.38*±0.06	7.79±0.52

Table 2: Error rates for CIFAR-10, CIFAR-100, SVHN, and STL-10.

shapes of a ship and a plane with its wings removed and the backgrounds of sky and sea are visually similar. In contrast, *our STOCO separates these ambiguous classes* in the feature space (cf. Fig. 3d), demonstrating that our method can learn more discriminative features.

4.3 Comparison with the State-of-the-art

We compare the proposed STOCO with the state-of-the-art methods on CIFAR-10 in Table 2, where results of existing methods are quoted from their respective papers or [44]. With 400 labels per class, all compared methods show small differences in performance; nevertheless, by combining SSL techniques, FixMatch greatly improves over Π-Model, Pseudo-Labeling, and Mean Teacher that are on their own; notably, our STOCO achieves the best result of 3.86%. With 25 labels per class, the methods based on technique combination are far ahead of those on their own by a margin larger than 20%, showing the huge advantages of technique combination; again, our method outperforms all the compared ones. With only 4 labels per class, the simpler FixMatch that combines self-training and consistency regularization is superior to the excellent ReMixMatch, which additionally integrates self-supervised learning and mixup; notably, with multiple stochastic classifiers, our STOCO produces a much better result than FixMatch and is on par with CoMatch, showing that our method is suitable for application scenarios with extremely scarce labels. Furthermore, we find that our method achieves a high classification accuracy of 7.17% on the challenging 40-label setting, which is only 2.40% and 3.31% lower than that on the 250- and 4000-label settings respectively, indicating that the benefit from increasing the number of labeled samples is limited on CIFAR-10 due to its simplicity.

The comparisons between different methods on the difficult CIFAR-100 are shown in Table 2, where most of the phenomena are similar to those on CIFAR-10. Besides, we emphasize the following several observations. **(1)** Compared to the results on CIFAR-10, these on CIFAR-100 still have a large room of improvement since the 100 classes in CIFAR-100 come from a fine-grained classification of 20 superclasses and thus are difficult to distinguish. **(2)** ReMixMatch performs better than FixMatch, especially on the hardest 400-label setting, which is due

to its use of distribution alignment (empirically found by [44]). **(3)** Our STOCO improves over FixMatch by a large margin, e.g., 7.40% on the 400-label setting, demonstrating the effectiveness of our method in tackling different learning scenarios with varying label conditions. Note that recent methods achieve the SSL goal of performing better with less supervision by technique combination; differently, our method gets closer to the goal via the use of a sample selection criterion based on stochastic consensus.

The results on SVHN are reported in Table 2, from which we take similar observations to those above. It is noteworthy that with only 4 labels per class, our STOCO outperforms the state-of-the-art FlexMatch by 2.51%, confirming the superiority of our method. When increasing the number of labels in each class from 4 to 100, we find that the performance gain is small (0.47%). This suggests that for a simple task like SVHN, a few labels are enough to get a good classification model. Although the same number of labels are available for each class, the results on CIFAR-100 are much worse than those on SVHN, implying that the required number of labeled samples to achieve good performance is task-dependent. Establishing a principled metric is expected to estimate this number in practical applications, such that the manual labeling efforts can be reduced.

We also organize the results on STL-10 in Table 2. With 100 labeled samples per class involved in training, the SSL methods based on technique combination exhibit clear advantages over others in such a challenging test; in particular, our STOCO is comparable to the state-of-the-art ones.

5 Conclusion

Semi-supervised learning (SSL) is a popular field, which aims to reduce the labeling cost in cases requiring domain expertise, e.g., medical diagnosis and cultural relic identification. Recent SSL methods focus on integrating various SSL techniques including self-training, where the criterion for selecting unlabeled samples with pseudo labels plays an important role in the empirical success. However, we note that the research direction of sample selection criterion is under-explored in SSL. To this end, we propose a novel criterion based on consistency among multiple stochastic classifiers, termed Stochastic Consensus (STOCO), which can be readily applied to any self-training based SSL framework. We choose the representative FixMatch as the baseline and achieve the state of the art on typical SSL benchmarks, especially in label-scarce cases. STOCO improves the model's generalization ability without losing simplicity, which helps audience expansion in the academic community and industrial deployment in recognition systems.

Acknowledgments. This work is supported in part by Program for Guangdong Introducing Innovative and Enterpreneurial Teams (No.: 2017ZT07X183), National Natural Science Foundation of China (No.: 61771201), and Guangdong R&D key project of China (No.: 2019B010155001). Correspondence to Kui Jia (email: kuijia@scut.edu.cn).

References

1. Angluin, D., Laird, P.: Learning from noisy examples. Mach. Learn. **2**, 343–370 (1988)
2. Assran, M., Caron, M., Misra, I., Bojanowski, P., Joulin, A., Ballas, N., Rabbat, M.: Semi-supervised learning of visual features by non-parametrically predicting view assignments with support samples. In: ICCV. pp. 8443–8452 (October 2021)
3. Balcan, M.F., Blum, A., Yang, K.: Co-training and expansion: Towards bridging theory and practice. In: NeurIPS. p. 89–96 (2004)
4. Berthelot, D., Carlini, N., Cubuk, E.D., Kurakin, A., Sohn, K., Zhang, H., Raffel, C.: Remixmatch: Semi-supervised learning with distribution matching and augmentation anchoring. In: ICLR (2020)
5. Berthelot, D., Carlini, N., Goodfellow, I., Papernot, N., Oliver, A., Raffel, C.A.: Mixmatch: A holistic approach to semi-supervised learning. In: NeurIPS. vol. 32 (2019)
6. Blum, A., Mitchell, T.: Combining labeled and unlabeled data with co-training. In: COLT. p. 92–100 (1998)
7. d'Alché Buc, F., Grandvalet, Y., Ambroise, C.: Semi-supervised marginboost. In: NeurIPS. p. 553–560 (2001)
8. Chen, M., Weinberger, K.Q., Blitzer, J.: Co-training for domain adaptation. In: NeurIPS. vol. 24 (2011)
9. Chen, Z., Zhuang, J., Liang, X., Lin, L.: Blending-target domain adaptation by adversarial meta-adaptation networks. In: CVPR. pp. 2243–2252 (2019)
10. Coates, A., Ng, A., Lee, H.: An analysis of single-layer networks in unsupervised feature learning. In: Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics. pp. 215–223 (2011)
11. Cubuk, E.D., Zoph, B., Shlens, J., Le, Q.: Randaugment: Practical automated data augmentation with a reduced search space. In: NeurIPS. vol. 33, pp. 18613–18624 (2020)
12. Dietterich, T.G.: Ensemble methods in machine learning. In: MCS. pp. 1–15 (2000)
13. Dizaji, K.G., Herandi, A., Deng, C., Cai, W., Huang, H.: Deep clustering via joint convolutional autoencoder embedding and relative entropy minimization. In: ICCV. pp. 5747–5756 (2017)
14. van Engelen, J., Hoos, H.: A survey on semi-supervised learning. Mach. Learn. **109**, 373–440 (2020)
15. Fei-Fei, L., Fergus, R., Perona, P.: One-shot learning of object categories. IEEE TPAMI **28**, 594–611 (2006)
16. French, G., Mackiewicz, M., Fisher, M.: Self-ensembling for visual domain adaptation. In: International Conference on Learning Representations (2018), `https://openreview.net/forum?id=rkpoTaxA-`
17. Gal, Y., Ghahramani, Z.: Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In: Balcan, M.F., Weinberger, K.Q. (eds.) Proc. Int. Conf. Mach. Learn. Proceedings of Machine Learning Research, vol. 48, pp. 1050–1059. PMLR, New York, New York, USA (20–22 Jun 2016), `https://proceedings.mlr.press/v48/gal16.html`
18. Grandvalet, Y., Bengio, Y.: Semi-supervised learning by entropy minimization. In: NeurIPS. pp. 529–536 (2004)
19. Guo, L.Z., Zhang, Z.Y., Jiang, Y., Li, Y.F., Zhou, Z.H.: Safe deep semi-supervised learning for unseen-class unlabeled data. In: III, H.D., Singh, A. (eds.) Proc. Int. Conf. Mach. Learn. Proceedings of Machine Learning Research, vol. 119, pp. 3897–3906. PMLR (13-18 Jul 2020)

20. Jabi, M., Pedersoli, M., Mitiche, A., Ayed, I.B.: Deep clustering: On the link between discriminative models and k-means. IEEE TPAMI **43**, 1887–1896 (2021)
21. Karim, M.R., Beyan, O., Zappa, A., Costa, I.G., Rebholz-Schuhmann, D., Cochez, M., Decker, S.: Deep learning-based clustering approaches for bioinformatics. Briefings in Bioinformatics **22**, 393–415 (2020)
22. Kingma, D., Welling, M.: Auto-encoding variational bayes. In: ICLR (2014)
23. Krizhevsky, A.: Learning multiple layers of features from tiny images. In: Technical report (2009)
24. Laine, S., Aila, T.: Temporal ensembling for semi-supervised learning. In: ICLR (2016)
25. Lee, D.H.: Pseudo-label : The simple and efficient semi-supervised learning method for deep neural networks. In: Proc. Int. Conf. Mach. Learn. Worksh. (07 2013)
26. Li, J., Xiong, C., Hoi, S.C.: Comatch: Semi-supervised learning with contrastive graph regularization. In: ICCV. pp. 9475–9484 (October 2021)
27. Li, W., Foo, C., Bilen, H.: Learning to impute: A general framework for semi-supervised learning. CoRR **abs/1912.10364** (2019), `http://arxiv.org/abs/1912.10364`
28. Liang, J., Yang, J., Lee, H.Y., Wang, K., Yang, M.H.: Sub-gan: An unsupervised generative model via subspaces. In: ECCV. pp. 698–714 (2018)
29. Lu, Z., Yang, Y., Zhu, X., Liu, C., Song, Y.Z., Xiang, T.: Stochastic classifiers for unsupervised domain adaptation. In: CVPR. pp. 9108–9117 (2020)
30. Luo, Y., Zhu, J., Li, M., Ren, Y., Zhang, B.: Smooth neighbors on teacher graphs for semi-supervised learning. In: CVPR. pp. 8896–8905 (2018)
31. van der Maaten, L., Hinton, G.: Visualizing data using t-sne. Journ. of Mach. Learn. Res. **9**, 2579–2605 (2008)
32. McLachlan, G.J.: Iterative reclassification procedure for constructing an asymptotically optimal rule of allocation in discriminant analysis. Journal of the American Statistical Association **70**, 365–369 (1975)
33. Miyato, T., Maeda, S.I., Koyama, M., Ishii, S.: Virtual adversarial training: A regularization method for supervised and semi-supervised learning. IEEE TPAMI **41**, 1979–1993 (2019)
34. Mousavi, S.M., Zhu, W., Ellsworth, W., Beroza, G.: Unsupervised clustering of seismic signals using deep convolutional autoencoders. IEEE Geoscience and Remote Sensing Letters **16**, 1693–1697 (2019)
35. Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B., Ng, A.Y.: Reading digits in natural images with unsupervised feature learning. In: Workshop of Proc. Neur. Info. Proc. Sys. (2011)
36. Pham, H., Dai, Z., Xie, Q., Le, Q.V.: Meta pseudo labels. In: CVPR. pp. 11557–11568 (June 2021)
37. Quinlan, J.R.: Miniboosting decision trees (1999)
38. Rasmus, A., Valpola, H., Honkala, M., Berglund, M., Raiko, T.: Semi-supervised learning with ladder networks. In: NeurIPS. p. 3546–3554 (2015)
39. Rizve, M.N., Duarte, K., Rawat, Y.S., Shah, M.: In defense of pseudo-labeling: An uncertainty-aware pseudo-label selection framework for semi-supervised learning. In: ICLR (2021), `https://openreview.net/forum?id=-ODN6SbiUU`
40. Rosenberg, C., Hebert, M., Schneiderman, H.: Semi-supervised self-training of object detection models. In: Seventh IEEE Workshops on Applications of Computer Vision. vol. 1, pp. 29–36 (2005)
41. Saito, K., Ushiku, Y., Harada, T.: Asymmetric tri-training for unsupervised domain adaptation. In: Proc. Int. Conf. Mach. Learn. p. 2988–2997 (2017)

42. Sajjadi, M., Javanmardi, M., Tasdizen, T.: Regularization with stochastic transformations and perturbations for deep semi-supervised learning. In: NeurIPS. vol. 29 (2016)
43. Scudder, H.: Probability of error of some adaptive pattern-recognition machines. IEEE Transactions on Information Theory **11**, 363–371 (1965)
44. Sohn, K., Berthelot, D., Carlini, N., Zhang, Z., Zhang, H., Raffel, C.A., Cubuk, E.D., Kurakin, A., Li, C.L.: Fixmatch: Simplifying semi-supervised learning with consistency and confidence. In: NeurIPS. vol. 33, pp. 596–608 (2020)
45. Tang, H., Chen, K., Jia, K.: Unsupervised domain adaptation via structurally regularized deep clustering. In: CVPR. pp. 8725–8735 (2020)
46. Tarvainen, A., Valpola, H.: Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In: NeurIPS. vol. 30 (2017)
47. Wang, Y., Guo, J., Song, S., Huang, G.: Meta-semi: A meta-learning approach for semi-supervised learning. CoRR **abs/2007.02394** (2020), `https://arxiv.org/abs/2007.02394`
48. Wei, C., Shen, K., ning Chen, Y., Ma, T.: Theoretical analysis of self-training with deep networks on unlabeled data. In: ICLR (2021)
49. Xie, J., Girshick, R., Farhadi, A.: Unsupervised deep embedding for clustering analysis. In: Proc. Int. Conf. Mach. Learn. pp. 478–487 (2016)
50. Xie, Q., Dai, Z., Hovy, E., Luong, T., Le, Q.: Unsupervised data augmentation for consistency training. In: NeurIPS. vol. 33, pp. 6256–6268 (2020)
51. Xie, Q., Luong, M.T., Hovy, E., Le, Q.V.: Self-training with noisy student improves imagenet classification. In: CVPR. pp. 10687–10698 (2020)
52. Zagoruyko, S., Komodakis, N.: Wide residual networks. In: BMVC (2016)
53. Zhang, B., Wang, Y., Hou, W., Wu, H., Wang, J., Okumura, M., Shinozaki, T.: Flexmatch: Boosting semi-supervised learning with curriculum pseudo labeling. In: Beygelzimer, A., Dauphin, Y., Liang, P., Vaughan, J.W. (eds.) NeurIPS (2021), `https://openreview.net/forum?id=3qMwV98zLIk`
54. Zhang, L., Qi, G.J.: Wcp: Worst-case perturbations for semi-supervised deep learning. In: CVPR. pp. 3911–3920 (2020)
55. Zhou, Z.H., Li, M.: Tri-training: exploiting unlabeled data using three classifiers. IEEE Trans. Knowl. Data Eng. **17**, 1529–1541 (2005)
56. Zou, Y., Yu, Z., Liu, X., Kumar, B.V.K.V., Wang, J.: Confidence regularized self-training. In: ICCV. pp. 5981–5990 (2019)
57. Zou, Y., Yu, Z., Vijaya Kumar, B.V.K., Wang, J.: Unsupervised domain adaptation for semantic segmentation via class-balanced self-training. In: ECCV. pp. 297–313 (2018)