How to Train the Teacher Model for Effective Knowledge Distillation

Shayan Mohajer Hamidi*[1], Xizhen Deng*[2], Renhao Tan[1], Linfeng Ye[1], and Ahmed Hussein Salamah[1]

[1] University of Waterloo, Waterloo ON N2L 3G1, Canada
{smohajer,cameron.tan,l44ye,ahamsalamah}@uwaterloo.ca
[2] University of Michigan, Ann Arbor, MI 48109, USA xizhen@umich.edu
* Authors contributed equally

Abstract. Recently, it was shown that the role of the teacher in knowledge distillation (KD) is to provide the student with an estimate of the true Bayes conditional probability density (BCPD). Notably, the new findings propose that the student's error rate can be upper-bounded by the mean squared error (MSE) between the teacher's output and BCPD. Consequently, to enhance KD efficacy, the teacher should be trained such that its output is close to BCPD in MSE sense. This paper elucidates that training the teacher model with MSE loss equates to minimizing the MSE between its output and BCPD, aligning with its core responsibility of providing the student with a BCPD estimate closely resembling it in MSE terms. In this respect, through a comprehensive set of experiments, we demonstrate that substituting the conventional teacher trained with cross-entropy loss with one trained using MSE loss in state-of-the-art KD methods consistently boosts the student's accuracy, resulting in improvements of up to 2.6%. The code for this paper is publicly available at: https://github.com/ECCV2024MSE/ECCV_MSE_Teacher.

Keywords: Knowledge distillation · Bayes conditional probability density · Mean squared error

1 Introduction

Knowledge distillation (KD), as introduced by [5] and popularized by [17], has emerged as a highly effective model compression technique, and has received significant attention from both academia and industry in recent years. At its core, KD entails the process of transferring the knowledge of a cumbersome model (teacher) into a lightweight counterpart (student). After the pioneering work by [17], numerous researchers have attempted to improve the performance of KD [3, 30, 36], and to understand why distillation works [2, 14, 16, 26, 28, 34, 52, 52].

An aspect of KD that has received relatively limited attention is the training of the teacher model. In most of the existing KD methods, the teacher is typically trained to optimize its own performance, despite the fact that such optimization does not necessarily translate into enhanced student performance [13, 27]. Hence,

to effectively train a student to attain high performance, it is crucial to align the training of the teacher accordingly.

Recently, [26] showed that the teacher's soft predictions can act as a proxy for the unknown true Bayes conditional probability distribution (BCPD) of label y given an input $\boldsymbol{x}$. Specifically, a teacher model trained using conventional cross-entropy (CE) loss function approximates the true BCPD of the underlying dataset [21], and then it passes this estimate to the student model. As such, the enhancement in the student's accuracy within KD stems from the fact that the student utilizes the teacher's BCPD approximation to train its own model. Additionally, [26] noted that as the teacher's predictions approach the BCPD, the generalization error of the student model decreases. More importantly, [14] showed that the classification error rate of the student is directly bounded by the MSE between the teacher's output and the BCPD, as established through the Rademacher analysis. This fact was further empirically confirmed by [35] where they treated the teacher's output as a supervisory signal for the student model. Hence, based on these findings, it is imperative for effective KD that the teacher is trained such that its output is close to the BCPD in the MSE sense.

Now the question is that how the teacher model should be trained to ensure that its prediction is close to the BCPD in the MSE sense? To answer to this question, in this paper, we prove that training a DNN via MSE (resp. CE) loss is equivalent to training it to minimize the expected MSE (resp. CE) between its output and the true BCPD. Hence, based on the discussions above, for an effective KD, the teacher should be trained using MSE loss function. We shall note that proximity in terms of MSE does not necessarily equate to proximity in terms of CE, and vice versa. Therefore, while the output of a teacher trained using CE loss serves as an estimate of the true BCPD, it may not necessarily be close to the BCPD in terms of MSE, which is essential for effective KD. In fact, we empirically show that although the student's accuracy is almost inversely proportional to the MSE between BCPD and teacher's output, such relationship does not exist between the student's accuracy and CE between BCPD and teacher's output.

Based on the discussions above, we claim that for an effective KD, the teacher should be trained using MSE loss. To demonstrate the effectiveness of the teacher trained via MSE loss in KD, we conduct a thorough set of experiments over CIFAR-100 and ImageNet datasets, and show that by **solely** replacing a conventional teacher trained with CE loss by one trained with MSE loss in the existing state-of-the-art KD methods, the student's accuracy consistently increases. We shall emphasize the fact that such gain is obtained without making any changes over the underlying KD methods such as its distillation loss function or any hyper-parameters. Additionally, we observe a slight decrease in the teacher's performance when trained using MSE loss, confirming that optimizing the teacher's performance is not necessary for an effective knowledge distillation process. To summarize, the contributions of this paper are as follows

- We introduce a theorem to show that training a DNN to minimize CE/MSE loss is equivalent to training it to minimize the CE/MSE of its output to the true BCPD.

- We show that for an effective KD, the student should be provided with an estimate of BCPD that is close to it in MSE sense; thus, as per our theorem, the teacher as a BCPD estimator should be trained via MSE loss.
- We conduct a thorough set of experiments over CIFAR-100 and ImageNet datasets, and show that by replacing the teacher trained by CE with the one trained by MSE in the existing state-of-the-art KD methods, the student's accuracy consistently increases.

2 Related works

2.1 Knowledge distillation

The concept of knowledge transfer, as a means of compression, was first introduced by [5]. Then, [17] popularized this concept by softening the teacher's and student's logits using temperature technique where the student mimics the soft probabilities of the teacher, and referred to it as KD. To improve the effectiveness of distillation, various forms of knowledge transfer methods have been introduced which could be mainly categorized into three types: (i) logit-based [4, 6, 23, 39, 55, 57], (ii) representation-based [36, 49, 53, 54], and (iii) relationship-based [25, 30, 33, 50].

2.2 Training a customized teacher for KD

In the literature, only a few works trained teachers specifically tailored for KD. [48] attempted to train a tolerant teacher which provides more secondary information to the student. They realized this via adding an extra term to facilitating a few secondary classes to emerge to complement the primary class. [13] regularized the teacher utilizing early-stopping during the training. Nevertheless, achieving optimal results may necessitate a thorough hyperparameter search, as the epoch number for identifying the best checkpoint can be particularly sensitive to various training settings, such as the learning rate schedule. Additionally, it is feasible to save multiple early teacher checkpoints, allowing the student to be distilled from them sequentially [20].

In addition, [45] stated that a checkpoint in the middle of the training procedure, often serves as a better teacher compared to the fully converged model. The authors in [15] used Lipschitz regularization so that the teacher can can learn the label distribution of the underlying dataset. Also [52] trained the teacher to have high conditional mutual information so that it can better predict the true BCPD. [40] trained the teachers to have more dispersed soft probabilities.

2.3 Training DNNs using MSE

Mean squared error (MSE) loss serves as a prevalent choice for training DNNs, particularly in the context of regression tasks. This loss function is widely embraced when the objective is to predict continuous values, and its formulation involves calculating the average of the squared differences between predicted

and actual values. Despite its established efficacy in regression scenarios, the landscape of loss functions in the realm of DNNs is vast and dynamic.

In contemporary practices, DNNs dedicated to classification tasks predominantly leverage the CE loss function. This method has gained substantial empirical favor, often surpassing MSE in the context of classification-oriented objectives. However, the empirical superiority of CE over MSE remains a topic of ongoing exploration. Notably, the existing body of literature does not uniformly advocate for a distinct advantage of CE in all scenarios.

Recent insights, as highlighted by the study conducted by [18], challenge the prevailing notion by showcasing that models trained with MSE not only hold their ground against their CE-trained counterparts across a diverse spectrum of tasks and settings but, intriguingly, exhibit superior classification performance in the majority of experimental conditions. These findings prompt a reevaluation of the perceived hierarchy between MSE and CE, underscoring the need for nuanced considerations when selecting the most suitable loss function based on the specificities of the task at hand. In light of such empirical observations, the applicability and performance of MSE in DNN training extend beyond the traditional confines of regression, warranting a more comprehensive exploration of its utility across various domains and applications.

3 Notation and Preliminaries

3.1 Notation

For a positive integer C, let $[C] \triangleq \{1, \ldots, C\}$. We use bold lowercase letters (e.g., $\boldsymbol{p}$) to represent vectors. Denote by $\boldsymbol{p}[i]$ the i-th element of vector $\boldsymbol{p}$. Also, $\{\boldsymbol{p}[c]\}_{c\in\mathcal{C}}$ is the set of all components of $\boldsymbol{p}$ with indices from the set $\mathcal{C}$. For two vectors $\boldsymbol{u}$ and $\boldsymbol{v}$, denote by $\boldsymbol{u}\cdot\boldsymbol{v}$ their inner product. We use $|\mathcal{C}|$ to denote the cardinality of a set $\mathcal{C}$. The transpose operation is denoted by $(\cdot)^{\mathrm{T}}$.

The cross-entropy of two probability distributions $\boldsymbol{p}_1$ and $\boldsymbol{p}_2$ is defined as $H(\boldsymbol{p}_1, \boldsymbol{p}_2) = \sum_{c=1}^{C} -\boldsymbol{p}_1[c] \log \boldsymbol{p}_2[c]$. For a random variable x, denote by $\mathbb{P}_x$ its probability distribution, and by $\mathbb{E}_x[\cdot]$ the expected value operation w.r.t. x. For two random variables x and y, denote by $\mathbb{P}_{(x,y)}$ their joint distribution.

3.2 True risk Vs. empirical risk

In a classification task with C classes, a DNN could be regarded as a mapping $f_{\boldsymbol{\theta}} : \boldsymbol{x} \to \boldsymbol{p}_{\boldsymbol{x}}$, where $\boldsymbol{\theta}$ represents all the model parameters, $\boldsymbol{x} \in \mathbb{R}^d$ is an input image, and $\boldsymbol{p}_{\boldsymbol{x}} \in \Delta^C$, where Δ^C is the C dimensional probability simplex. Then, the classifier predicts the correct label of $\boldsymbol{x}$, denoted by y, as $\hat{y} = \arg\max_{c\in[C]} \boldsymbol{p}_{\boldsymbol{x}}[c]$. As such, the error rate of f is defined as $\epsilon = \Pr\{\hat{y} \neq y\}$, and its accuracy is equal to $1 - \epsilon$. One may learn such a classifier by minimizing the *true* risk

$$\begin{aligned} R(f_{\boldsymbol{\theta}}, \ell) &\triangleq \mathbb{E}_{(\boldsymbol{x},y)} \left[\ell\left(y, \boldsymbol{p}_{\boldsymbol{x}}\right)\right] = \mathbb{E}_{\boldsymbol{x}} \left[\mathbb{E}_{y|\boldsymbol{x}} \left[\ell\left(y, \boldsymbol{p}_{\boldsymbol{x}}\right)\right]\right] \\ &= \mathbb{E}_{\boldsymbol{x}} \left[\left(\boldsymbol{p}^*_{\boldsymbol{x}}\right)^{\mathrm{T}} \cdot \boldsymbol{\ell}(\boldsymbol{p}_{\boldsymbol{x}})\right], \end{aligned} \tag{1}$$

where $\ell(\cdot)$ is the loss function and $\boldsymbol{\ell}(\cdot) \triangleq [\ell(1,\cdot),\ldots,\ell(C,\cdot)]$ is the vector of loss function, and $\boldsymbol{p}^*_{\boldsymbol{x}} \triangleq [\Pr(y|\boldsymbol{x})]_{y\in[C]}$ is Bayes class probability distribution over the labels, i.e, the BCPD.

However, in a typical deep learning algorithm, both the probability density function of $\boldsymbol{x}$, namely $\mathbb{P}_{\boldsymbol{x}}$, and also $\boldsymbol{p}^*_{\boldsymbol{x}}$ are unknown. Hence, one may learn such a classifier by instead minimizing the *empirical* risk on a training sample $\mathcal{D} \triangleq \{(\boldsymbol{x}_n, y_n)\}_{n=1}^N$ defined as:

$$R_{\text{emp}}(f_{\boldsymbol{\theta}}, \ell) \triangleq \frac{1}{N} \sum_{n\in[N]} \mathbf{y}_n^{\mathrm{T}} \cdot \boldsymbol{\ell}\left(\boldsymbol{p}_{\boldsymbol{x}_n}\right), \tag{2}$$

where $\mathbf{y}_n^{\mathrm{T}}$ is the one-hot vector with its y_n-th entry set to one and all other entries set to zero. By comparing Eq. (1) and Eq. (2), we see that in Eq. (2): (i) $\mathbb{P}_{\boldsymbol{x}}$ is approximated by $\frac{1}{N}$, and (ii) $\boldsymbol{p}^*_{\boldsymbol{x}}$ is approximated by $\mathbf{y}$ which is an unbiased estimation of $\boldsymbol{p}^*_{\boldsymbol{x}}$. The former assumption is reasonable, however, the latter results in a significant loss in granularity. To delve deeper into this matter, it is crucial to recognize that images inherently carry a wealth of information, and the practice of assigning a one-hot vector to $\boldsymbol{p}^*_{\boldsymbol{x}}$ tends to lead to a significant loss of this information. As we will explore further in the subsequent subsection, KD emerges as a strategy that, to some extent, alleviates this issue, providing a mechanism to better transfer the nuanced information embedded within images.

3.3 Estimating BCPD by the teacher in KD

In KD, the role of the teacher is to provide the student with a better estimate of $\boldsymbol{p}^*_{\boldsymbol{x}}$ compared to one-hot vectors. To elucidate, denote by $\boldsymbol{p}^t_{\boldsymbol{x}}$ and $\boldsymbol{p}^s_{\boldsymbol{x}}$ the pre-trained teacher's and student's outputs to sample $\boldsymbol{x}$, respectively. Then, the student uses the teacher's estimate of $\boldsymbol{p}^*_{\boldsymbol{x}}$, and minimizes

$$R_{\text{kd}}(f_{\boldsymbol{\theta}}, \ell) \triangleq \frac{1}{N} \sum_{n\in[N]} \left(\boldsymbol{p}^t_{\boldsymbol{x}_n}\right)^{\mathrm{T}} \cdot \boldsymbol{\ell}\left(\boldsymbol{p}^s_{\boldsymbol{x}_n}\right). \tag{3}$$

Note that the one-hot vector $\mathbf{y}_n$ in Eq. (2) is now replaced by the teacher's output probability $\boldsymbol{p}^t_{\boldsymbol{x}_n}$ in Eq. (3). In fact, the effectiveness of KD lies in the fact that $\boldsymbol{p}^t_{\boldsymbol{x}_n}$ serves as a better estimate of $\boldsymbol{p}^*_{\boldsymbol{x}}$ compared to the one-hot vector $\mathbf{y}_n^{\mathrm{T}}$.

3.4 Student's generalization error and accuracy

In this subsection, our objective is to identify the key characteristics that the estimated BCPD should possess in order to enhance the accuracy of the student.

As shown by [26], the student's generalization error is upper-bounded as

$$\mathbb{E}\left[(R_{\text{kd}}(f_{\boldsymbol{\theta}},\ell) - R(f_{\boldsymbol{\theta}},\ell))^2\right] \leq \frac{1}{N}\text{Var}\left[(\boldsymbol{p}^t_{\boldsymbol{x}})^{\mathrm{T}} \cdot \boldsymbol{\ell}(\boldsymbol{p}_{\boldsymbol{x}})\right] + \kappa\left(\mathbb{E}\left[\|\boldsymbol{p}^t_{\boldsymbol{x}} - \boldsymbol{p}^*_{\boldsymbol{x}}\|\right]\right)^2, \tag{4}$$

where κ is a positive constant number. When N is large, which is commonly the case for datasets in existing literature, the second term will dominate the

right-hand side of Eq. (4). This implies that smaller average $\|\boldsymbol{p}_{\boldsymbol{x}}^t - \boldsymbol{p}_{\boldsymbol{x}}^*\|_2$ will lead to $R_{\text{emp}}(f_{\boldsymbol{\theta}})$ being a better approximation of the true risk $R(f_{\boldsymbol{\theta}})$, minimizing it should then lead to a better learned model.

On the other hand, [35] empirically showed that the accuracy of the student is almost inversely proportional to $\|\boldsymbol{p}_{\boldsymbol{x}}^t - \boldsymbol{p}_{\boldsymbol{x}}^*\|$. In addition, [14] showed that the accuracy of the student is directly bounded by the MSE between teacher's prediction and BCPD through the Rademacher analysis.

Therefore, the quality of the estimates provided by the teacher to the student can be measured by the MSE between its output and to the true $\boldsymbol{p}_{\boldsymbol{x}}^*$. Based on this, in the next section, we show that for an effective KD, the teacher should be indeed trained via MSE loss, and not CE loss.

4 Methodology

In this section, first in Sec. 4.1, we introduce a theorem demonstrating that training a DNN model with MSE (CE) loss function results in minimizing the MSE (CE) between its output and the BCPD. Then, in Sec. 4.2, we use a synthetic dataset to empirically validate the introduced theorem, and to show that (i) closeness in MSE sense does not necessarily mean closeness in CE sense, and (ii) for an effective KD the teacher's output should be close to the true BCPD in MSE sense. Then, we conclude that the teacher should be trained via MSE loss function.

4.1 MSE loss Vs. CE loss

In a classification task with C classes, it has been shown that the risk in Eq. (1), for $\ell = \{\text{CE}, \text{MSE}\}$, is minimized when $\boldsymbol{p}_{\boldsymbol{x}} = \boldsymbol{p}_{\boldsymbol{x}}^*$ [21]. However, since the underlying $\mathbb{P}_{(\boldsymbol{x},y)}$ is unknown, the DNNs are trained to minimize the empirical risk in Eq. (2), and consequently they can only approximate the true BCPD[3].

Hence, a teacher trained by either CE or MSE can approximate the true BCPD. However, it is crucial to note that these two estimates differ, as precisely established in the following theorem.

Theorem 1. *For* $\ell = \{CE, MSE\}$

$$\min_{\theta} \mathbb{E}_{(\boldsymbol{x},y)} \left[\ell\left(y, \boldsymbol{p}_{\boldsymbol{x}}\right)\right] \equiv \min_{\theta} \mathbb{E}_{(\boldsymbol{x},y)} \left[\ell\left(\boldsymbol{p}_{\boldsymbol{x}}^*, \boldsymbol{p}_{\boldsymbol{x}}\right)\right]. \tag{5}$$

Proof. Please refer to the *Supplementary materials.*

Theorem 1 implies that when minimizing the expected loss, the resulting model attempts to generate outputs that closely approximate those of an "ideal" model. More importantly, this degree of closeness is quantified by the loss function. Specifically, (i) if $\ell = \text{MSE}$ then a model trained to minimize the expected squared

[3] Aside from the fact that $\mathbb{P}_{(\boldsymbol{x},y)}$ is unknown, training cannot typically find the global minimum hindering the DNNs to give accurate BCPD.

error between y and $\boldsymbol{p}_{\boldsymbol{x}}$ will generate outputs that minimize the expected squared error between $\boldsymbol{p}^*_{\boldsymbol{x}}$ and $\boldsymbol{p}_{\boldsymbol{x}}$; and (ii) if $\ell = \text{CE}$ then a model trained to minimize the expected cross-entropy between y and $\boldsymbol{p}_{\boldsymbol{x}}$ will generate outputs that minimize the expected cross-entropy between $\boldsymbol{p}^*_{\boldsymbol{x}}$ and $\boldsymbol{p}_{\boldsymbol{x}}$.

Thus, the BCPD estimates provided by the teachers trained by CE loss and MSE loss are different; in that, the former estimate is close to the true BCPD in CE sense, also the latter estimate is close to the true BCPD in MSE sense. In the next subsection, we empirically show that for the student to have high accuracy, the teacher's estimate should be close to the true BCPD in MSE sense, rather than in CE sense.

4.2 MSE proximity Vs. CE proximity

In this subsection, our intention is two-fold: (i) we aim to empirically show that although the student's accuracy is almost inversely proportional to the MSE between BCPD and teacher's output (as also demonstrated by [14, 35]), such relationship does not exist between the student's accuracy and CE between BCPD and teacher's output; and (ii) to empirically validate Theorem 1. Toward this aim, since the true BCPD is unknown for the popular datasets in the literature, we generate a synthetic dataset whose BCPD is known.

• **Generating dataset**: Inspired from [35], we generate a 3-class toy Gaussian dataset with 10^5 data points. The dataset is divided into training, validation, and test sets with a split ratio $[0.9, 0.05, 0.05]$.

The sampling process is implemented as follows: we first choose the label y using a uniform distribution across all the 3 classes. Next, we sample $x|_{y=k} \sim \mathcal{N}\left(\mu_k, \sigma^2 I\right)$ as the input signal. Here, μ_k is a 30-dim vector with entries randomly selected from $\{-\delta_\mu, 0, \delta_\mu\}$. Then, we calculate the BCPD of the samples using the fact that $p^*(y|\boldsymbol{x}) \propto p(\boldsymbol{x}|y)p(y)$. Particularly, as y follows uniform distribution, we have $\boldsymbol{p}^*(y|\boldsymbol{x}) = \frac{\boldsymbol{p}(\boldsymbol{x}|y=k)}{\sum_{j\neq k} p(\boldsymbol{x}|y=j)}$. Following $\boldsymbol{p}(\boldsymbol{x}|y=k) \sim \mathcal{N}(\mu_k, \sigma^2 I)$, we find $\boldsymbol{p}^*(y|\boldsymbol{x})$ should have a Softmax form as

$$\boldsymbol{p}^*(y=k|\boldsymbol{x}) = \frac{\exp(s_k)}{\sum_{j\neq k}\exp(s_j)}; \quad s_i = -\frac{1}{2\sigma^2}\|\boldsymbol{x}-\mu_i\|_2^2. \tag{6}$$

• **Training setup**: The underlying model (for both teacher and student) is a 3-layer MLP with ReLU activation function, and the hidden size is 128 for each layer. We set the learning rate as 5×10^{-4}, the batch size as 32, and the number of training epochs is 100. In our experiments, we set $\sigma = 4$ and $\delta_\mu = 1$.

Now, we conduct two sets of experiments as elaborated in the sequel.

Set 1: the student's accuracy as a function of MSE($\boldsymbol{p}^*_{\boldsymbol{x}}$, $\boldsymbol{p}_{\boldsymbol{x}}$) and CE($\boldsymbol{p}^*_{\boldsymbol{x}}$, $\boldsymbol{p}_{\boldsymbol{x}}$). Here, we conduct some experiments to understand which type of proximity, i.e., whether proximity in CE sense or MSE sense, is better for the sake of the student accuracy. To this end, we generate 100 perturbed/noisy versions of $\boldsymbol{p}^*_{\boldsymbol{x}}$ by adding some random noise to it, and denote any such perturbed version by $\tilde{\boldsymbol{p}}^*_{\boldsymbol{x}}$.

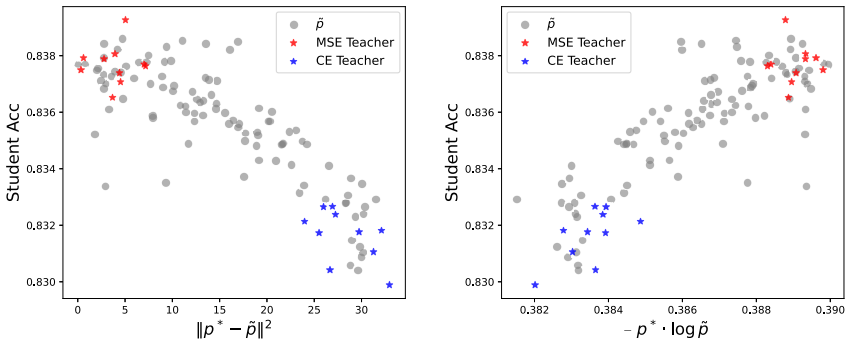

Fig. 1: Student accuracy as a function of (left) the MSE between $\boldsymbol{p}_{\boldsymbol{x}}$ and $\tilde{\boldsymbol{p}}_{\boldsymbol{x}}$; and (right) CE between $\boldsymbol{p}_{\boldsymbol{x}}$ and $\tilde{\boldsymbol{p}}_{\boldsymbol{x}}$. The gray dots are noisy versions of the true $\boldsymbol{p}_{\boldsymbol{x}}$. Also, the red and blue dots represent the points corresponding to the estimates provided by MSE and CE teachers, respectively.

Then, we train the student 100 times, each time using one of the noisy $\tilde{\boldsymbol{p}}^*_{\boldsymbol{x}}$, i.e., each time the student is trained via loss Eq. (3) with $\boldsymbol{p}^t_{\boldsymbol{x}_n}$ replaced by $\tilde{\boldsymbol{p}}^*_{\boldsymbol{x}}$. The resulting student's accuracy corresponding to these 100 noisy BCPD is depicted in Fig. 1, where in the left and right figure we set x-axis to the MSE and CE between the $\boldsymbol{p}^*_{\boldsymbol{x}}$ and $\tilde{\boldsymbol{p}}^*_{\boldsymbol{x}}$, respectively (the points corresponding to noisy $\boldsymbol{p}^*_{\boldsymbol{x}}$ are denoted by gray dots).

As seen in Fig. 1, the student's accuracy is almost inversely proportional to the MSE between $\boldsymbol{p}^*_{\boldsymbol{x}}$ and $\tilde{\boldsymbol{p}}^*_{\boldsymbol{x}}$. Consequently, the teacher can enhance the student's performance by providing it with an estimate of $\boldsymbol{p}^*_{\boldsymbol{x}}$ that is closer to it in MSE sense. On the other hand, as seen in the right figure depicted in Fig. 1, such relationship does not exist; in that, minimizing the CE between $\boldsymbol{p}^*_{\boldsymbol{x}}$ and $\tilde{\boldsymbol{p}}^*_{\boldsymbol{x}}$ does not necessarily increase the student's performance.

Set 2: empirically validating Theorem 1. The previous set of experiments showed that the student's accuracy can be improved if it is given a BCPD estimate which is close to $\boldsymbol{p}^*_{\boldsymbol{x}}$ in MSE sense. Here, we aim to empirically validate the Theorem 1, and therefore based on our discussions above, the teacher should be trained via MSE loss.

Hence, we train the teacher 10 times using MSE loss, and 10 times using CE loss. We denote the teacher's estimate of $\boldsymbol{p}^*_{\boldsymbol{x}}$ by $\tilde{\boldsymbol{p}}^*_{\boldsymbol{x},\mathrm{MSE}}$ and $\tilde{\boldsymbol{p}}^*_{\boldsymbol{x},\mathrm{CE}}$ when it is trained by CE and MSE losses, respectively.

Now, once again we train the student using these two types of estimate (10 estimates for each type) and record the student's accuracy. The results are depicted in Fig. 1, where we used red and blue dots to show the points corresponding to $\tilde{\boldsymbol{p}}^*_{\boldsymbol{x},\mathrm{MSE}}$ and $\tilde{\boldsymbol{p}}^*_{\boldsymbol{x},\mathrm{CE}}$ estimates, respectively.

Observing the two figures we can conclude that (i) training the teacher using MSE/CE loss yields the BCPD estimate which is close to $\boldsymbol{p}^*_{\boldsymbol{x}}$ in MSE/CE sense;

(ii) the closeness in CE and MSE sense is rather different; and (iii) the teacher trained via MSE loss results in a better performance for the student model.

Lastly, we provide a toy example to understand why proximity in MSE sense is different from that in CE sense. Assume that the true BCPD is $\boldsymbol{p}^* = [0.3, 0.7]$. Consider the following two estimates of the BCPD.

- Estimate 1, $E_1 = [0.29, 0.71]$. Here $\text{MSE}(\boldsymbol{p}^*, E_1) = 0.01$ and $\text{CE}(\boldsymbol{p}^*, E_1) = 0.610$.
- Estimate 2, $E_2 = [0.2, 0.8]$. Here $\text{MSE}(\boldsymbol{p}^*, E_2) = 0.10$ and $\text{CE}(\boldsymbol{p}^*, E_2) = 0.51$.

In the CE sense, E_2 is a superior estimator of $\boldsymbol{p}^*$. However, from the standpoint of MSE, E_1 is a more accurate representation of $\boldsymbol{p}^*$. Consequently, if a DNN is trained using the CE loss, it tends to produce an output akin to E_2 which is at a relatively greater distance from the true distribution $\boldsymbol{p}^*$ in the MSE sense.

5 Experiments

We conclude the paper with extensive experiments demonstrating the superior effectiveness of MSE teacher compared to CE teacher in different state-of-the-art KD methods. The outcomes of these experiments collectively contribute to a compelling argument for the preferential utilization of MSE teachers in the KD variants.

• **Terminology**: Hereafter, we refer to teachers trained by the CE and MSE losses as the “CE teacher" and “MSE teacher", respectively.

• **Organization**: The experiments are organized as follows: in Sec. 5.1 and Sec. 5.2, we conduct experiments on CIFAR-100 dataset; also Sec. 5.3 provides experiments on ImageNet dataset. In Sec. 5.4, we compare MSE and CE teachers in the semi-supervised distillation task. Lastly, we evaluate the performance of MSE teacher in binary classification knowledge distillation task in Sec. 5.5.

• **Plug-and-play nature of MSE teacher**: In all the experiments conducted in this section, when evaluating the performance of the MSE teacher, we refrain from tuning any hyper-parameters in the underlying knowledge transfer methods, i.e., all hyper-parameters remain the same as those used in the corresponding benchmark methods. We present the classification accuracies on the test set for both the teacher model and its distilled student model.

In addition, we should note that for training the MSE teacher, we use the same training setup as that used for the CE loss. One may further improve our results by further tuning the training hyper-parameters. In addition, similarly to the conventional KD methods, we use the same teacher for different students, without adjusting the teacher based on the specifics of each student model.

Table 1: The test accuracy (%) of student networks on CIFAR-100 (averaged over 5 runs), with teacher-student pairs of the same/different architectures. The subscript denotes the improvement achieved by replacing CE teacher with MSE teacher. We use **bold** numbers and asterisk (*) to denote the best results and to identify the results reproduced on our local machines, respectively.

Teachers and students with the **same** architectures.												
Teacher	ResNet-56		ResNet-110		ResNet-110		WRN-40-2		WRN-40-2		VGG-13	
	CE	MSE	CE	MSE	CE	MSE	CE	MSE	CE	MSE	CE	MSE
Accuracy	72.34	72.14	74.31	73.43	74.31	73.43	75.61	74.99	75.61	74.99	74.64	73.20
Student	ResNet-20		ResNet-20		ResNet-32		WRN-16-2		WRN-40-1		VGG-8	
Accuracy	69.06		69.06		71.14		73.26		71.98		70.36	
AT	70.55	70.80 +0.25	70.22	70.58 +0.36	72.31	73.50 +1.19	74.08	74.30 +0.22	72.77	73.05 +0.28	71.43	71.72 +0.29
PKT	70.34	70.84 +0.50	70.25	70.55 +0.30	72.61	72.90 +0.29	74.54	74.85 +0.31	73.45	74.10 +0.65	72.88	73.10 +0.47
SP	69.67	70.77 +1.10	70.04	70.75 +0.71	72.69	73.34 +0.65	73.83	74.60 +0.77	72.43	73.30 +0.87	72.68	73.19 +0.61
CC	69.63	69.99 +0.36	69.48	69.89 +0.41	71.48	71.75 +0.27	73.56	73.87 +0.31	72.21	72.50 +0.29	70.71	71.00 +0.29
RKD	69.61	70.50 +0.89	69.25	70.20 +0.95	71.82	72.62 +0.80	73.35	73.66 +0.31	72.22	72.67 +0.45	71.48	71.88 +0.40
VID	70.38	70.61 +0.23	70.16	70.49 +0.33	72.61	73.05 +0.44	74.11	74.44 +0.33	73.30	73.58 +0.28	71.23	71.57 +0.34
CRD	71.16	71.43 +0.27	71.46	71.87 +0.41	73.48	74.03 +0.55	75.48	75.85 +0.37	74.14	74.86 +0.72	73.94	74.25 +0.31
REVIEW	71.89	72.15 +0.26	71.65*	72.04 +0.39	73.89	74.02 +0.13	76.12	76.29 +0.17	75.09	75.33 +0.24	74.84	74.90 +0.06
DKD	71.97	72.27 +0.30	71.51*	71.88 +0.37	74.11	74.32 +0.21	76.24	76.76 +0.52	74.81	75.53 +0.72	74.68	74.92 +0.24
HSAKD	72.58	**72.74** +0.16	72.64*	**73.07** +0.43	74.97*	**75.52** +0.55	77.20	**77.55** +0.35	77.00	**77.32** +0.32	75.42*	**75.76** +0.34
Teachers and students with **different** architectures.												
Teacher	ResNet-50		ResNet-50		ResNet-32×4		ResNet-32×4		WRN-40-2		VGG-13	
	CE	MSE	CE	MSE	CE	MSE	CE	MSE	CE	MSE	CE	MSE
Accuracy	79.34	74.54	79.34	74.54	79.41	75.24	79.41	75.24	75.61	74.99	74.64	73.20
Student	MobileNetV2		VGG-8		ShuffleNetV1		ShuffleNetV2		ShuffleNetV1		MobileNetV2	
Accuracy	64.60		70.36		70.50		71.82		70.50		64.60	
AT	58.58	59.63 +1.05	71.84	72.12 +0.28	71.73	72.06 +0.33	72.73	74.11 +1.38	73.32	74.33 +1.01	59.40	62.07 +2.67
PKT	66.52	67.02 +0.50	73.10	73.45 +0.35	74.10	74.81 +0.71	74.69	76.34 +1.65	73.89	75.39 +1.50	67.13	68.08 +0.95
SP	68.08	69.00 +0.92	73.34	74.04 +0.70	73.48	74.57 +1.09	74.56	75.70 +1.14	74.52	75.72 +1.20	66.30	67.03 +0.73
CC	65.43	65.90 +0.47	70.25	70.90 +0.65	71.14	71.77 +0.63	71.29	73.02 +1.73	71.38	71.80 +0.42	64.86	65.05 +0.19
RKD	64.43	64.88 +0.45	71.50	72.05 +0.55	72.28	73.19 +0.91	73.21	73.62 +0.41	72.21	73.25 +1.04	64.52	65.32 +0.80
VID	67.57	67.77 +0.20	70.30	70.55 +0.25	73.38	73.89 +0.51	73.40	74.67 +1.27	73.61	75.03 +1.42	65.56	65.82 +0.26
CRD	69.11	69.15 +0.04	74.30	74.55 +0.25	75.11	75.81 +0.70	75.65	76.54 +0.89	76.05	76.43 +0.38	69.70	69.77 +0.07
REVIEW	69.89	70.03 +0.14	73.43*	73.90 +0.47	77.45	77.78 +0.33	77.78	78.81 +0.03	77.14	77.25 +0.11	70.37	70.81 +0.44
DKD	70.35	71.40 +1.05	73.94*	75.14 +1.17	76.45	77.15 +0.70	77.07	77.52 +0.45	76.70	77.30 +0.60	69.71	70.22 +0.51
HSAKD	71.83*	**72.67** +0.84	75.87*	**76.34** +0.47	79.51*	**79.81** +0.30	79.93	**80.09** +0.17	78.51	**78.82** +0.31	71.09*	**72.40** +1.31

5.1 CIFAR-100

This dataset comprises 50,000 training and 10,000 test color images, each of size 32×32, and is annotated for 100 classes [22].

• **Teacher-student pairs:** In line with the configurations of CRD [41], we use a couple of teacher-student pairs with identical and different network architectures for our experiments (see Tab. 1). We conduct each experiment over 5 independent runs and report the average accuracy (for the accuracy variances, refer to the *Supplementary materials*).

• **KD variants:** For comprehensive comparisons, we compare using a MSE teacher Vs. CE teacher in the existing state-of-the-art distillation methods, including KD [17], AT [54], PKT [32], SP [42], CC [33], RKD [31], VID [1], CRD [41], DKD [55], REVIEWKD [7], and HSAKD [49].

• **Training setup:** For all variants of knowledge distillation and settings in this paper, SGD is applied as the optimizer. We train the student for 240 epochs for all experiments with an initial learning rate of 0.05 by default, which will be decayed by factor of 0.1 at epoch 150, 180, 210. For MobileNetV2, ShuffleNetV1, and ShuffleNetV2, a smaller initial learning rate of 0.01 is used. We adopt batch size of 64. In addition, we report the hyper-parameters used for underlying KD variants in *Supplementary materials*.

Table 2: Additional experiments on CIFAR-100, when the feature based distillation methods are combined with conventional KD method. The test accuracy (%) of student networks on CIFAR-100 (averaged over 5 runs), with teacher-student pairs of the same/different architectures. The subscript denotes the improvement achieved by replacing CE teacher with MSE teacher.

Teacher	ResNet-110		WRN-40-2		VGG-13		ResNet-50		ResNet-32×4		VGG-13	
	CE	MSE	CE	MSE	CE	MSE	CE	MSE	CE	MSE	CE	MSE
Accuracy	74.31	73.43	75.61	74.99	74.64	73.20	79.34	74.54	79.41	75.24	74.64	73.20
Student	ResNet-20		WRN-16-2		VGG-8		VGG-8		ShuffleNetV2		MobileNetV2	
Accuracy	69.06		73.26		70.36		71.14		73.26		64.60	
AT+KD	70.97	71.34 +0.37	75.32	75.65 +0.33	73.48	73.89 +0.41	74.01	74.22 +0.21	75.39	77.63 +2.24	65.13	66.76 +1.63
PKT+KD	70.72	71.15 +0.43	75.33	75.73 +0.40	73.25	73.50 +0.25	73.61	73.80 +0.19	74.66	76.13 +1.47	68.13	68.89 +0.76
SP+KD	71.02	71.40 +0.38	74.98	75.63 +0.65	73.49	73.80 +0.31	73.52	73.97 +0.45	74.88	76.86 +1.98	68.41	69.37 +0.96
CC+KD	70.88	71.31 +0.43	75.09	75.48 +0.39	73.04	73.46 +0.42	73.48	70.79 +0.31	74.71	76.29 +1.58	68.02	68.39 +0.37
RKD+KD	70.77	71.48 +0.71	74.89	75.61 +0.52	72.97	73.30 +0.33	73.51	73.86 +0.35	74.55	75.82 +1.27	67.87	68.38 +0.51
VID+KD	71.10	71.53 +0.43	75.14	75.62 +0.48	73.19	73.47 +0.28	73.46	73.88 +0.42	74.85	75.97 +1.12	68.27	68.75 +0.48
CRD+KD	71.56	71.95 +0.39	75.64	75.85 +0.21	74.29	74.60 +0.31	74.58	74.98 +0.40	76.05	76.93 +0.88	69.94	70.23 +0.29
REVIEW+KD	71.78*	72.01 +0.23	76.22*	76.35 +0.13	74.96*	75.07 +0.11	73.89*	74.44 +0.55	77.89*	78.10 +0.21	71.05*	71.86 +0.81

• **Results:** The results are reported in Tab. 1, where the upper-part of the table comprises the feature-based methods that are combined with KD, and the lower-part contains three logit-based KD variants.

By noting the results in Tab. 1, the following observations could be made:

- Substituting the CE teacher with the MSE teacher in the benchmark KD methods consistently leads to an enhancement in the student's performance. This improvement can reach up to 2.67%.
- The improvement achieved in teacher-student pairs with different architectures is notably more substantial (these pairs are the last three columns of Tab. 1).
- The accuracy of the teacher experiences a slight decline when employing the MSE loss. This observation underscores the distinction between training the teacher model for KD and training it solely for optimizing its individual performance. It affirms that these are distinct tasks, each with its own set of considerations and trade-offs.

5.2 Additional Experiments on CIFAR-100

In this subsection, we follow the CRD paper [41], and combine the feature based distillation methods with conventional KD for achieving a higher performance. The results are listed in Tab. 2. As seen, the MSE teacher consistently yield a better accuracy compared to its CE counterpart.

Table 3: Top-1 and Top-5 student's test accuracy (%) on ImageNet validation set for 4 different KD methods using CE and MSE teachers (RN and MN stand for ResNet and MobileNet, respectively). **Bold** numbers and asterisk (*) to denote the best results and to identify the results reproduced on our local machines, respectively.

Teacher-Student	Teacher Performance		KD		DKD		REVIEW + KD		CRD +KD	
	Top1	Top5	Top1	Top5	Top1	Top5	Top1	Top5	Top1	Top5
RN34-R18	CE 73.31	91.42	71.03	90.05	71.70	90.41	71.84*	90.77*	71.38	90.49
	MSE 71.66	90.83	71.58	90.77	71.93	91.23	**72.16**	90.93	71.57	90.36
RN50-MNV2	CE 76.13	92.86	70.50	89.80	72.05	91.05	72.56*	91.00*	71.37*	90.41*
	MSE 73.88	90.97	70.92	90.08	72.34	91.17	**72.91**	92.38	71.58	90.62

5.3 ImageNet

ImageNet [37] is a large-scale dataset used in visual recognition tasks, containing around 1.2 million training and 50K validation images.

• **Teacher-student pairs:** Following the settings of [41, 51], we use 2 popular teacher-student pairs for our experiments (see Tab. 3).

• **Training setup:** We set the initial learning rate to 0.1 and divide the learning rate by 10 at 30, 60, and 90 epochs. We follow the standard training process but train for 20 more epochs (*i.e.*, 120 epochs in total). Weight decay is set to 0.0001.
• **Results:** We note that across all the knowledge transfer methods reported in Tab. 3, replacing the CE teacher by MSE teacher consistently leads to an increase in the student accuracy. For example, when considering teacher-student pairs ResNet34-ResNet18 and ResNet50-MobileNetV2, the increase in the student's Top-1 accuracy in KD is 0.55% and 0.42%, respectively.

It is important to highlight that achieving such gains over the ImageNet dataset is considered substantial. Taken together, the results obtained across both CIFAR-100 and ImageNet datasets underscore the effectiveness of opting for MSE teachers over their CE counterparts.

5.4 MSE teacher in semi-supervised distillation

Semi-supervised learning [8, 9, 11, 12, 19, 24, 46, 56] is a popular technique due to its ability to generate pseudo-labels for larger unlabeled dataset. In the context of KD, the teacher model has the responsibility of generating pseudo-labels for new, unlabeled examples (beyond its traditional function of providing soft targets during training).

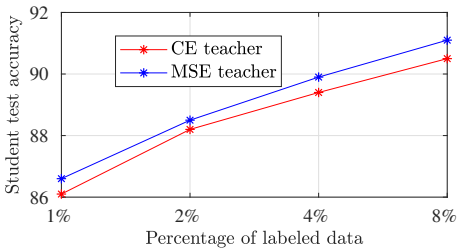

Fig. 2: The student's accuracy in semi supervised distillation for CE and MSE teachers.

To assess the MSE teacher's performance in a semi-supervised learning scenario, we conducted experiments using the CIFAR-10 dataset, following the settings outlined in [8], with the student model being ResNet18. In this experimental setup, although the dataset consists of 50,000 training images, only a small percentage of them are labeled—specifically, 1% (500), 2% (1000), 4% (2000), and 8% (4000). The outcomes are visualized in Fig. 2, where the student accuracy is plotted against the number of labeled samples. The results are averaged over three independent runs. As observed, the results demonstrate that not only can the MSE teacher effectively perform in a semi-supervised distillation scenario, but it also surpasses the performance of the CE teacher.

5.5 Binary classification on customized CIFAR-{10, 100}

The common understanding is that the enhancement in the accuracy of a student model in binary classification tends to be more modest compared to that observed in multi-class classification scenarios. This discrepancy arises due to the inherent limitations on the amount of information transferred from the teacher to the student network in binary classification settings, as documented in several studies [10, 29, 38, 43, 44, 47].

In this section, we want to empirically verify the effectiveness of MSE teacher in binary classification tasks. To this end, we create three binary classification datasets from CIFAR-$\{10, 100\}$ datasets as explained in the sequel.

• **Dataset 1:** Following a similar approach to that described in [29], we construct the CIFAR-2×5 dataset, wherein the input distribution exhibits a "sub-classes" structure. Specifically, we merge the first 5 classes of CIFAR-10 to form class one, while the remaining 5 classes constitute class two.

• **Dataset 2:** We keep the training/testing samples from only two classes in the CIFAR-100 dataset: those belonging to class 20 and class 40, creating a dataset referred to as CIFAR-20-40.

• **Dataset 3:** Similar to dataset 2, but we keep class 50 and 70 from CIFAR-100, which we refer to as CIFAR-50-70.

Now, we use VGG-13 and MobileNetv2-0.1 as the teacher and student models, respectively; and use conventional KD, AT, CC and VID to distill knowledge from CE and MSE teachers to the student.

The results for all three tasks are summarized in Tab. 4, with the reported values representing the average across five distinct runs. As seen, the MSE teacher yields a better student's accuracy all the cases. Also, it is worth noting that in some cases, for instance, AT over CIFAR-2×5 dataset, the distillation method hurts the accuracy of the student.

Table 4: The student's accuracy (%) in binary classification knowledge distillation on variants of CIFAR dataset. The results are averaged over five runs.

Dataset	CIFAR-2×5		CIFAR-26-45		CIFAR-50-74	
Student Acc.	76.94		65.50		66.30	
Teacher	CE	MSE	CE	MSE	CE	MSE
KD	77.00	77.34	69.70	70.15	71.40	71.23
AT	67.39	68.21	64.70	65.33	67.10	69.12
CC	77.19	77.64	70.30	70.83	69.70	71.38
VID	77.16	77.47	69.25	69.88	69.80	69.97

6 Conclusion

This paper elucidated the significance of training the teacher model with MSE loss, which effectively minimizes the MSE between its output and BCPD. This approach aligns with the core responsibility of the teacher, namely, providing the student with a BCPD estimate that closely resembles it in terms of MSE. Through a comprehensive series of experiments, we demonstrated the efficacy of substituting the conventional teacher trained with CE loss with one trained using MSE loss in state-of-the-art KD methods. Notably, this substitution consistently enhanced the student's accuracy, leading to improvements of up to 2.6%. In addition, we empirically showed the superior performance of MSE teacher in semi-supervised distillation task.

References

1. Ahn, S., Hu, S., Damianou, A., Lawrence, N., Dai, Z.: Variational information distillation for knowledge transfer pp. 9155–9163 (06 2019). https://doi.org/10.1109/CVPR.2019.00938
2. Allen-Zhu, Z., Li, Y.: Towards understanding ensemble, knowledge distillation and self-distillation in deep learning. arXiv preprint arXiv:2012.09816 (2020)
3. Anil, R., Pereyra, G., Passos, A., Ormandi, R., Dahl, G.E., Hinton, G.E.: Large scale distributed neural network training through online distillation. arXiv preprint arXiv:1804.03235 (2018)
4. Beyer, L., Zhai, X., Royer, A., Markeeva, L., Anil, R., Kolesnikov, A.: Knowledge distillation: A good teacher is patient and consistent. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10925–10934 (2022)
5. Buciluǎ, C., Caruana, R., Niculescu-Mizil, A.: Model compression. In: Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining. pp. 535–541 (2006)
6. Chen, D., Mei, J.P., Wang, C., Feng, Y., Chen, C.: Online knowledge distillation with diverse peers. In: Proceedings of the AAAI Conference on Artificial Intelligence. pp. 3430–3437 (2020)
7. Chen, P., Liu, S., Zhao, H., Jia, J.: Distilling knowledge via knowledge review. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5008–5017 (2021)
8. Chen, T., Kornblith, S., Swersky, K., Norouzi, M., Hinton, G.E.: Big self-supervised models are strong semi-supervised learners. Advances in neural information processing systems **33**, 22243–22255 (2020)
9. Chi, Z., Gu, L., Liu, H., Wang, Y., Yu, Y., Tang, J.: Metafscil: A meta-learning approach for few-shot class incremental learning. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 14166–14175 (2022)
10. Chi, Z., Gu, L., Zhong, T., Liu, H., YU, Y., Plataniotis, K.N., Wang, Y.: Adapting to distribution shift by visual domain prompt generation. In: The Twelfth International Conference on Learning Representations
11. Chi, Z., Mohammadi Nasiri, R., Liu, Z., Lu, J., Tang, J., Plataniotis, K.N.: All at once: Temporally adaptive multi-frame interpolation with advanced motion modeling. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVII 16. pp. 107–123. Springer (2020)
12. Chi, Z., Wang, Y., Yu, Y., Tang, J.: Test-time fast adaptation for dynamic scene deblurring via meta-auxiliary learning. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 9137–9146 (2021)
13. Cho, J.H., Hariharan, B.: On the efficacy of knowledge distillation. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 4794–4802 (2019)
14. Dao, T., Kamath, G.M., Syrgkanis, V., Mackey, L.: Knowledge distillation as semiparametric inference. In: International Conference on Learning Representations (2020)
15. Dong, C., Liu, L., Shang, J.: Toward student-oriented teacher network training for knowledge distillation. In: The Twelfth International Conference on Learning Representations (2023)
16. Hamidi, S.M.: Training neural networks on remote edge devices for unseen class classification. IEEE Signal Processing Letters **31**, 1004–1008 (2024). https://doi.org/10.1109/LSP.2024.3383948

17. Hinton, G., Vinyals, O., Dean, J.: Distilling the knowledge in a neural network. arXiv preprint arXiv:1503.02531 (2015)
18. Hui, L., Belkin, M.: Evaluation of neural architectures trained with square loss vs cross-entropy in classification tasks. In: International Conference on Learning Representations (2020)
19. Iliopoulos, F., Kontonis, V., Baykal, C., Menghani, G., Trinh, K., Vee, E.: Weighted distillation with unlabeled examples. Advances in Neural Information Processing Systems **35**, 7024–7037 (2022)
20. Jin, X., Peng, B., Wu, Y., Liu, Y., Liu, J., Liang, D., Yan, J., Hu, X.: Knowledge distillation via route constrained optimization. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 1345–1354 (2019)
21. Kanaya, F., Miyake, S.: Bayes statistical behavior and valid generalization of pattern classifying neural networks. IEEE Transactions on Neural Networks **2**(4), 471–475 (1991)
22. Krizhevsky, A., Hinton, G., et al.: Learning multiple layers of features from tiny images (2009)
23. Li, Z., Huang, Y., Chen, D., Luo, T., Cai, N., Pan, Z.: Online knowledge distillation via multi-branch diversity enhancement. In: Proceedings of the Asian Conference on Computer Vision (2020)
24. Liu, H., Gu, L., Chi, Z., Wang, Y., Yu, Y., Chen, J., Tang, J.: Few-shot class-incremental learning via entropy-regularized data-free replay. In: European Conference on Computer Vision. pp. 146–162. Springer (2022)
25. Liu, Y., Cao, J., Li, B., Yuan, C., Hu, W., Li, Y., Duan, Y.: Knowledge distillation via instance relationship graph. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7096–7104 (2019)
26. Menon, A.K., Rawat, A.S., Reddi, S., Kim, S., Kumar, S.: A statistical perspective on distillation. In: International Conference on Machine Learning. pp. 7632–7642. PMLR (2021)
27. Mirzadeh, S.I., Farajtabar, M., Li, A., Levine, N., Matsukawa, A., Ghasemzadeh, H.: Improved knowledge distillation via teacher assistant. In: Proceedings of the AAAI conference on artificial intelligence. vol. 34, pp. 5191–5198 (2020)
28. Mobahi, H., Farajtabar, M., Bartlett, P.: Self-distillation amplifies regularization in hilbert space. Advances in Neural Information Processing Systems **33**, 3351–3361 (2020)
29. Müller, R., Kornblith, S., Hinton, G.: Subclass distillation. arXiv preprint arXiv:2002.03936 (2020)
30. Park, W., Kim, D., Lu, Y., Cho, M.: Relational knowledge distillation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3967–3976 (2019)
31. Park, W., Kim, D., Lu, Y., Cho, M.: Relational knowledge distillation pp. 3967–3976 (2019)
32. Passalis, N., Tefas, A.: Learning deep representations with probabilistic knowledge transfer (2018)
33. Peng, B., Jin, X., li, D., Zhou, S., Wu, Y., Liu, J., Zhang, Z., Liu, Y.: Correlation congruence for knowledge distillation pp. 5006–5015 (10 2019). https://doi.org/10.1109/ICCV.2019.00511
34. Phuong, M., Lampert, C.: Towards understanding knowledge distillation. In: International conference on machine learning. pp. 5142–5151. PMLR (2019)
35. Ren, Y., Guo, S., Sutherland, D.J.: Better supervisory signals by observing learning paths. In: International Conference on Learning Representations (2021)

36. Romero, A., Ballas, N., Kahou, S.E., Chassang, A., Gatta, C., Bengio, Y.: Fitnets: Hints for thin deep nets. arXiv preprint arXiv:1412.6550 (2014)
37. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al.: Imagenet large scale visual recognition challenge. International journal of computer vision **115**, 211–252 (2015)
38. Sajedi, A., Plataniotis, K.N.: On the efficiency of subclass knowledge distillation in classification tasks. arXiv preprint arXiv:2109.05587 (2021)
39. Stanton, S., Izmailov, P., Kirichenko, P., Alemi, A.A., Wilson, A.G.: Does knowledge distillation really work? Advances in Neural Information Processing Systems **34**, 6906–6919 (2021)
40. Tan, C., Liu, J.: Improving knowledge distillation with a customized teacher. IEEE Transactions on Neural Networks and Learning Systems (2022)
41. Tian, Y., Krishnan, D., Isola, P.: Contrastive representation distillation (2020)
42. Tung, F., Mori, G.: Similarity-preserving knowledge distillation. 2019 IEEE/CVF International Conference on Computer Vision (ICCV) pp. 1365–1374 (2019), https://api.semanticscholar.org/CorpusID:198179476
43. Tzelepi, M., Passalis, N., Tefas, A.: Efficient online subclass knowledge distillation for image classification. In: 2020 25th International Conference on Pattern Recognition (ICPR). pp. 1007–1014 (2021). https://doi.org/10.1109/ICPR48806.2021.9411995
44. Tzelepi, M., Passalis, N., Tefas, A.: Online subclass knowledge distillation. Expert Systems with Applications **181**, 115132 (2021)
45. Wang, C., Yang, Q., Huang, R., Song, S., Huang, G.: Efficient knowledge distillation from model checkpoints. Advances in Neural Information Processing Systems **35**, 607–619 (2022)
46. Wu, Y., Chi, Z., Wang, Y., Feng, S.: Metagcd: Learning to continually learn in generalized category discovery. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 1655–1665 (2023)
47. Wu, Y., Chi, Z., Wang, Y., Plataniotis, K.N., Feng, S.: Test-time domain adaptation by learning domain-aware batch normalization. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 38, pp. 15961–15969 (2024)
48. Yang, C., Xie, L., Qiao, S., Yuille, A.L.: Training deep neural networks in generations: A more tolerant teacher educates better students. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 33, pp. 5628–5635 (2019)
49. Yang, C., An, Z., Cai, L., Xu, Y.: Hierarchical self-supervised augmented knowledge distillation. arXiv preprint arXiv:2107.13715 (2021)
50. Yang, C., Zhou, H., An, Z., Jiang, X., Xu, Y., Zhang, Q.: Cross-image relational knowledge distillation for semantic segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12319–12328 (2022)
51. Yang, J., Martinez, B., Bulat, A., Tzimiropoulos, G.: Knowledge distillation via softmax regression representation learning. In: International Conference on Learning Representations (2020)
52. Ye, L., Hamidi, S.M., Tan, R., YANG, E.H.: Bayes conditional distribution estimation for knowledge distillation based on conditional mutual information. In: The Twelfth International Conference on Learning Representations (2024), https://openreview.net/forum?id=yV6wwEbtkR
53. Yim, J., Joo, D., Bae, J., Kim, J.: A gift from knowledge distillation: Fast optimization, network minimization and transfer learning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4133–4141 (2017)

54. Zagoruyko, S., Komodakis, N.: Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer (2017), https://arxiv.org/abs/1612.03928
55. Zhao, B., Cui, Q., Song, R., Qiu, Y., Liang, J.: Decoupled knowledge distillation. arXiv preprint arXiv:2203.08679 (2022)
56. Zhong, T., Chi, Z., Gu, L., Wang, Y., Yu, Y., Tang, J.: Meta-dmoe: Adapting to domain shift by meta-distillation from mixture-of-experts. Advances in Neural Information Processing Systems **35**, 22243–22257 (2022)
57. Zhu, X., Gong, S., et al.: Knowledge distillation by on-the-fly native ensemble. Advances in neural information processing systems **31** (2018)