Learning with noisy labels using low-dimensional model trajectory

Anonymous Author(s)
Affiliation
Address
email

Abstract

Recent work shows that deep neural networks (DNNs) first learn clean samples and then memorize noisy samples. Early stopping can therefore be used to improve performance when training with noisy labels. It was also shown recently that the training trajectory of DNNs can be approximated in a low-dimensional subspace using PCA. The DNNs can then be trained in this subspace achieving similar or better generalization. These two observations were utilized together, to further boost the generalization performance of vanilla early stopping on noisy label datasets. In this paper, we probe this finding further on different real-world and synthetic label noises. First, we show that the prior method is sensitive to the early stopping hyper-parameter. Second, we investigate the effectiveness of PCA, for approximating the optimization trajectory under noisy label information. We propose to estimate low-rank subspace through robust and structured variants of PCA, namely Robust PCA, and Sparse PCA. We find that the subspace estimated through these variants can be less sensitive to early stopping, and can outperform PCA to achieve better test error when trained on noisy labels.

1 Introduction

Deep neural networks have been successful in a wide variety of real-world tasks. However, they owe a major chunk of their success to large, carefully curated, and manually annotated datasets [7, 20]. In several applications, however, the annotations can be costly or difficult to obtain. Thus, several applications use unreliable annotation sources such as search engines, or crowd-sourcing [24, 22]. Thus, the *annotations/labels on training data may be noisy leading to a distribution shift at test time*.

Deep neural networks can easily memorize very large datasets [25], and they eventually memorize the noisy labels, leading to poor generalization. Several works have pointed out that deep neural networks tend to learn samples with clean labels early in training, and then memorize noisy labels during later stages [15, 19, 2]. This property has been leveraged in different ways to improve generalization performance when training labels are noisy.

The recent work of [12, 13] showed that neural networks can be trained in very low-dimensional subspaces while achieving similar or better generalization. They then utilize this property, in conjunction with early stopping to train on datasets with noisy labels. They first sample the model trajectory formed by gradient descent **and early stop** so the model has not yet fitted to the noisy labels. Then, they use *principal component analysis (PCA)* on the model trajectory to construct a low-dimensional subspace of the trajectory. Finally, they train a new network from initialization in the subspace. By leveraging early stopping and the low-dimensional optimization objective, they show an impressive generalization boost over vanilla early stopping.

Submitted to Workshop on Distribution Shifts, 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

However, it is unclear whether the success of the above method stems from the use of early stopping or due to the low-dimensional subspace for training the neural network. In many scenarios, the choice of early stopping may be unclear due to noisy validation data. Also, while early stopping is a useful defense against label noise recent work has also shown that real-world label noises and some synthetic label noises can be learned early adversely affecting generalization [18, 23, 26]. Intuitively, fitting random labels for DNNs should require a larger dimensional optimization trajectory [14]. Hence, restricting the optimization trajectory to be low-dimensional should provide a regularization against noisy labels. However, it is unclear whether PCA-based dimensionality reduction for the optimization trajectory is ideal for training with noisy labels.

In this work, we attempt to probe these questions. We first show that leveraging a low-dimensional model trajectory to regularize against noisy labels is fragile to early stopping. We then explore the different subspace estimation algorithms, namely **Robust-PCA and Sparse-PCA** to better regularize the recovered subspace. These variants have additional properties, which we discuss in detail below that may be useful for training with noisy labels. We conduct experiments for these PCA variants on different synthetic and real-world noisy variations of the CIFAR-10 dataset [10]. We find that while Robust-PCA does not always outperform PCA, Sparse-PCA is consistently less sensitive to early stopping and often outperforms PCA to achieve better generalization.

2 Background

For a deep neural network (DNN), we let $w \in \mathbb{R}^n$ denote its parameters. Let the parameter trajectory during regular training be denoted by $\{w_i^s\}_{i=0,1,\ldots,t}$, where w_0^s denotes initial parameters, and w_i^s denotes the parameters of DNN after a specific number of update iterations (usually an epoch). The dynamic linear dimensionality reduction (DLDR) algorithm proposed by [12] shows that neural networks can be trained in low-dimensional subspaces. The algorithms consist of two stages, sampling the subspace, and training the model on the sampled subspace. [12] show that neural networks can show equal or better test accuracy in the generated subspace for common datasets such as CIFAR-10 [10] and Imagenet [4] on a variety of common architectures. The algorithms are detailed as Algorithm 1 and 2.

Algorithm 1 DLDR Sampling

Sample parameter trajectory $\{w_0^s, w_1^s \ldots w_t^s\}$ along training;
$\bar{w} = \frac{1}{t}\sum_{i=1}^{t} w_i^s$;
$W = \{w_1^s - \bar{w}, w_2^s - \bar{w} \ldots w_t^s - \bar{w}\}$;
Perform SVD on $W^T W$ and truncate till d largest eigenvectors $\{v_1, v_2 \ldots v_d\}$ and eigenvalues $\{\sigma_1^2, \sigma_2^2 \ldots \sigma_d^2\}$ are obtained;
$u_i = \frac{1}{\sigma_i} W v_i$;
$P = [u_1, u_2 \ldots u_d]$;

Algorithm 2 Subspace Training

$k \leftarrow 0$;
$w_0 \leftarrow w_1^s$;
while not converged **do**
    Sample batch of data $\mathbb{B}_k$
    Compute gradient g_k on batch $\mathbb{B}_k$
    $w_{k+1} \leftarrow w_k - \alpha P P^T g_k$;   ▷ α denotes learning rate
    $k \leftarrow k + 1$;
end while

Intuitively, in order to fit random labels, the dimensionality of the subspace required should be larger. Thus, the DLDR algorithm controls the regularization by two mechanisms. First, sampling the subspace till an early epoch provides regularization, as the model learns clean labels in the early epochs [2, 15, 19]. Second, decreasing the dimensionality of the subspace provides an additional regularization, and reduces fitting to noisy labels. Thus, the early stop epoch and subspace dimensionality control the regularization, with these denoted by t and d, respectively. The prior work of [12] conducted experiments by synthetically creating corrupted CIFAR-10 labels, and using the above algorithm to show an impressive boost over vanilla SGD on clean test accuracy.

3 Proposed Method

By the Eckart-Young theorem, PCA provides optimal low-rank approximation by maximizing the Frobenius norm. As discussed, the DLDR framework uses SVD/PCA to create the low-rank subspace for optimization. For training with noisy labels, we instead propose alternative techniques for subspace estimation, namely **Robust-PCA** and **Sparse-PCA** to regularize the subspace estimate. While there exist multiple other variations of PCA with interesting properties, a detailed study of all these variants is beyond the scope of this paper. We leave further exploration of these variants as future work. We detail the advantages, Robust and Sparse-PCA have over PCA for training with noisy labels below.

Robust-PCA: Since PCA focuses on finding subspaces that maximize the variance of data, it is sensitive to the presence of outliers [21, 5, 8]. Robust-PCA instead is much less susceptible to sparse large outliers compared to PCA [11, 5]. For classification with noisy labels, gradients from the noisy data can be considered outliers, and PCA may over-emphasize them. Robust-PCA may therefore function better for training with noisy labels.

Sparse-PCA: Deep networks are usually over-parameterized allowing them to overfit to noisy labels [25]. A line of work has shown that only a few of these parameters are critical to generalization [6, 17]. Recent work also showed training only the critical parameters can improve training on noisy labels [19], which proposed to update a pre-defined fraction of the parameters that they selected as critical. These 'critical' parameters are based on a heuristic inspired by the Lottery Ticket Hypothesis [6]. In a similar essence, we propose to use Sparse-PCA to create the model trajectory. *Sparse-PCA functions similar to PCA with an additional constraint that the principal components should be sparse*. Thus, with Sparse-PCA, only a fraction of network weights can be updated, providing further regularization against noisy labels. The sparsity for each eigenvector is a hyper-parameter choice. Sparse-PCA also has an additional property of retaining consistency even when the number of samples is very few. PCA, however, is not consistent in this setting [16]. This property may be beneficial since DNNs have a very large number of parameters (in the order of millions), but the trajectory is approximated using very few samples (up to 100). Lastly, Sparse-PCA does not guarantee that different principal components are orthogonal (unlike PCA) without additional constraints. Since we only require the components to span a subspace, this property does not affect the algorithm.

There are multiple algorithms present in the literature for solving Robust-PCA and Sparse-PCA. For Robust-PCA, we use the SGD solver implementation by HyperSpy [3]. For Sparse-PCA, we use the OPIT solver proposed in [1]. Thus, compared to DLDR we only change the subspace estimation algorithm and use Robust-PCA and Sparse-PCA instead of vanilla PCA and do not modify Algorithm 2. We find that Sparse-PCA often works better than PCA, and can often outperform it while being less susceptible to the choice of early stopping.

4 Experiments

We evaluate our proposed approach on the CIFAR-10 dataset [10]. For synthetic noise, we randomly perturb a fraction of labels in the training set, consistent with existing literature. We discuss the different forms of label noises below:

1. **Symmetric** - This is a form of synthetic noise, where the noisy labels from every single class are uniformly split among all other classes.
2. **Pairflip** - In this synthetic noise, the noisy labels from each class are flipped into its adjacent class. This form of noise simulates noisy labels in fine-grained classification and is generally more easily learned during early epochs than symmetric noise [23].
3. **CIFAR10-N** - A collection of noisy human annotations of the CIFAR-10 training set [18]. We use the 'worst' subset of annotations, which takes a union of noisy labels across the dataset by 3 independent annotators. The noise level for CIFAR10-N 'worst' is around 40%. This type of noise is also learned easily during early epochs.

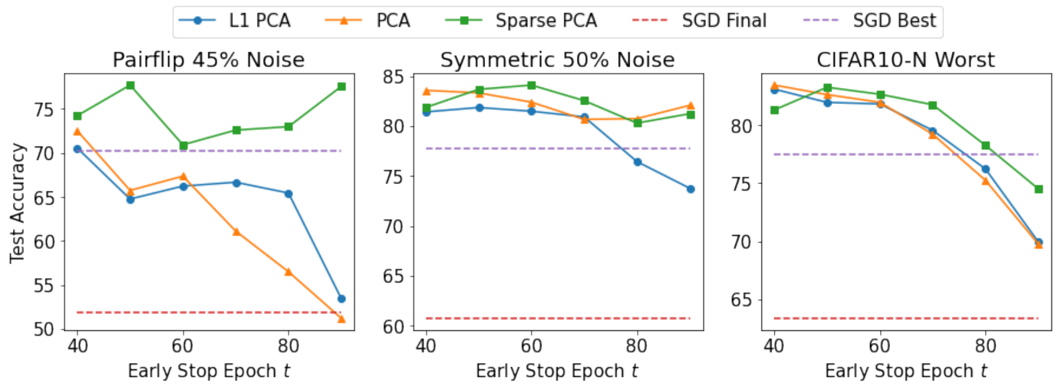

Figure 1: Comparison of different PCA variants across different synthetic and real label noises on CIFAR-10. Results are presented using PreActResnet-18, with subspace dimension kept as $d = 15$.

We evaluate the performance of all the models using the test split of CIFAR-10. We train a PreActResNet-18 [9] model with batch size 128 and use the common data augmentations, i.e., random crop with a padding of 4 pixels on each side, and horizontal flipping. For the first phase of training, while sampling the model checkpoints for subspace estimation, we use an SGD optimizer with 0.9 momentum, and weight decay of $5e-4$. We train for a total of 100 epochs with an initial learning rate of 0.1 and decay it by a factor of 10 at the 50th and 75th epochs. We sample checkpoints at every epoch for the subspace estimation. We use the same model checkpoints for PCA, Robust PCA, and Sparse-PCA for a fair comparison.

For the second phase of training, after the subspace is estimated, we train the network for 20 epochs projecting the gradient to the subspace after each iteration as shown in Algorithm 2. We set the initial learning rate to 1, and decay it by a factor of 10 at the 10th and 15th epochs. The learning rate can be set fairly high, due to subspace projection [12]. We use an SGD optimizer with 0.9 momentum and no weight decay. We experiment with different subspace early stop epoch t, and keep the subspace dimensionality $d = 15$ for all algorithms. We report additional experiments varying subspace dimension, d in Appendix A.1. For Sparse-PCA, we use a sparsity level of 90% for each eigenvector. For Robust PCA, we use default hyperparameters defined by HyperSpy. For PCA, we use the default implementation provided by the authors [12]. Figure 1 shows experimental results of different PCA variants on various types of label noises. We also show two baselines, SGD performance at the optimal early stop (SGD Best), and SGD final checkpoint performance.

We observe that for pairflip noise of 45%, Sparse-PCA can always outperform PCA and always obtains higher accuracy than SGD best accuracy. PCA however is extremely sensitive to early stopping and often performs even worse than optimal SGD early stop. Robust-PCA is slightly less sensitive to early-stopping than PCA for $t > 60$. For symmetric noise, Sparse-PCA does not clearly outperform PCA but shows similar or better performance when $t > 50$. Sparse-PCA also consistently performs better than SGD with optimal early stopping. Robust-PCA shows worse performance than PCA for symmetric noise. For the worst subset of CIFAR-10N annotations, Sparse PCA can outperform PCA when $t > 40$, and more consistently outperforms SGD with optimal early stopping. Robust-PCA shows similar performance to PCA, with no clear distinction. While none of the PCA variants consistently outperform PCA across all early-stopping thresholds, Sparse-PCA is often less sensitive to it. Sparse-PCA also achieves better generalization compared to PCA, on the challenging forms of label noise that are learned early, i.e., Pairflip and CIFAR10-N worst.

5 Conclusion

In this work, we probe how early stopping combined with learning in low-dimensional subspaces can improve generalization when training with noisy labels. We first show that the prior work on this topic is sensitive to the choice of early stopping, and may not offer much benefit for challenging forms of label noise that may be learned early. We then investigate the use of PCA variants to recover a low-dimensional subspace and find that Sparse-PCA often outperforms the prior method. We hope this work will open new theoretical and empirical studies on exploiting low-dimensional subspaces for noisy label training.

References

[1] Karim Abed-Meraim, Adel Hafiane, Nguyen Linh Trung, et al. Sparse subspace tracking in high dimensions. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5892–5896. IEEE, 2022.

[2] Yingbin Bai, Erkun Yang, Bo Han, Yanhua Yang, Jiatong Li, Yinian Mao, Gang Niu, and Tongliang Liu. Understanding and improving early stopping for learning with noisy labels. *Advances in Neural Information Processing Systems*, 34:24392–24403, 2021.

[3] Francisco De La Peña, Eric Prestat, Vidar Tonaas Fauske, Pierre Burdet, Jonas Lähnemann, Tom Furnival, Petras Jokubauskas, Magnus Nord, Tomas Ostasevicius, Katherine E MacArthur, et al. hyperspy/hyperspy: Release v1. 6.5. *Zenodo*, 2019.

[4] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[5] Jiashi Feng, Huan Xu, and Shuicheng Yan. Online robust pca via stochastic optimization. *Advances in neural information processing systems*, 26, 2013.

[6] Jonathan Frankle and Michael Carbin. The lottery ticket hypothesis: Finding sparse, trainable neural networks. In *International Conference on Learning Representations*, 2019.

[7] Bo Han, Jiangchao Yao, Gang Niu, Mingyuan Zhou, Ivor Tsang, Ya Zhang, and Masashi Sugiyama. Masking: A new perspective of noisy supervision. *Advances in neural information processing systems*, 31, 2018.

[8] Jun He, Laura Balzano, and Arthur Szlam. Incremental gradient on the grassmannian for online foreground and background separation in subsampled video. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1568–1575. IEEE, 2012.

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *European conference on computer vision*, pages 630–645. Springer, 2016.

[10] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[11] Nojun Kwak. Principal component analysis based on l1-norm maximization. *IEEE transactions on pattern analysis and machine intelligence*, 30(9):1672–1680, 2008.

[12] Tao Li, Lei Tan, Qinghua Tao, Yipeng Liu, and Xiaolin Huang. Low dimensional landscape hypothesis is true: Dnns can be trained in tiny subspaces. *arXiv preprint arXiv:2103.11154*, 2021.

[13] Tao Li, Yingwen Wu, Sizhe Chen, Kun Fang, and Xiaolin Huang. Subspace adversarial training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13409–13418, 2022.

[14] Fusheng Liu, Haizhao Yang, and Qianxiao Li. Short optimization paths lead to good generalization, 2022.

[15] Sheng Liu, Jonathan Niles-Weed, Narges Razavian, and Carlos Fernandez-Granda. Early-learning regularization prevents memorization of noisy labels. *Advances in neural information processing systems*, 33:20331–20342, 2020.

[16] Dan Shen, Haipeng Shen, and James Stephen Marron. Consistency of sparse pca in high dimension, low sample size contexts. *Journal of Multivariate Analysis*, 115:317–333, 2013.

[17] Yi-Lin Sung, Varun Nair, and Colin A Raffel. Training neural networks with fixed sparse masks. *Advances in Neural Information Processing Systems*, 34:24193–24205, 2021.

[18] Jiaheng Wei, Zhaowei Zhu, Hao Cheng, Tongliang Liu, Gang Niu, and Yang Liu. Learning with noisy labels revisited: A study using real-world human annotations. In *International Conference on Learning Representations*, 2022.

[19] Xiaobo Xia, Tongliang Liu, Bo Han, Chen Gong, Nannan Wang, Zongyuan Ge, and Yi Chang. Robust early-learning: Hindering the memorization of noisy labels. In *International Conference on Learning Representations*, 2021.

[20] Tong Xiao, Tian Xia, Yi Yang, Chang Huang, and Xiaogang Wang. Learning from massive noisy labeled data for image classification. In *CVPR*, 2015.

[21] Huan Xu, Constantine Caramanis, and Sujay Sanghavi. Robust pca via outlier pursuit. *Advances in neural information processing systems*, 23, 2010.

[22] Yan Yan, Rómer Rosales, Glenn Fung, Ramanathan Subramanian, and Jennifer Dy. Learning from multiple annotators with varying expertise. *Machine learning*, 95(3):291–327, 2014.

[23] Xingrui Yu, Bo Han, Jiangchao Yao, Gang Niu, Ivor Tsang, and Masashi Sugiyama. How does disagreement help generalization against label corruption? In *International Conference on Machine Learning*, pages 7164–7173. PMLR, 2019.

[24] Xiyu Yu, Tongliang Liu, Mingming Gong, and Dacheng Tao. Learning with biased complementary labels. In *Proceedings of the European conference on computer vision (ECCV)*, pages 68–83, 2018.

[25] Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning requires rethinking generalization. In *International Conference on Learning Representations*, 2017.

[26] Songzhu Zheng, Pengxiang Wu, Aman Goswami, Mayank Goswami, Dimitris Metaxas, and Chao Chen. Error-bounded correction of noisy labels. In *International Conference on Machine Learning*, pages 11447–11457. PMLR, 2020.

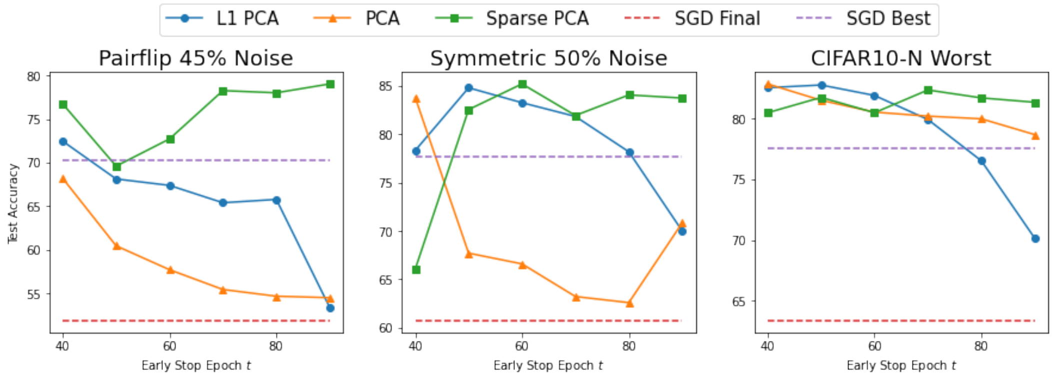

Figure 2: Comparison of PCA variants on noisy CIFAR-10. Subspace dimension, $d = 10$.

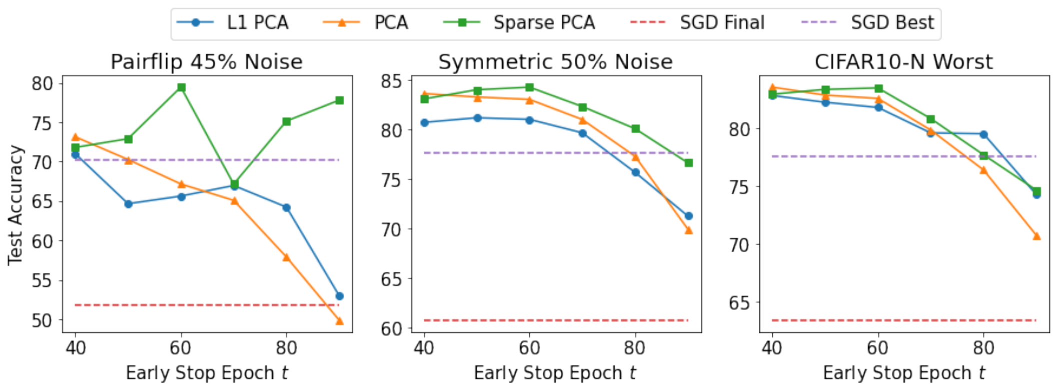

Figure 3: Comparison of PCA variants on noisy CIFAR-10. Subspace dimension, $d = 20$

A Appendix

A.1 Subspace Dimension

[12], relies on subspace dimension as a regularization mechanism, in addition to early stopping. Thus, in this section, we experiment with modifying the subspace dimension for all the PCA variants, to $d = 10$ and $d = 20$ as shown in Figure 2 and 3. We observe similar trends as discussed previously. Sparse PCA tends to be less susceptible to early stopping compared to PCA. Sparse PCA also still outperforms PCA across all the noisy datasets and obtains better generalization.