Backpropagation-Friendly Eigendecomposition

Wei Wang, Zheng Dang, Yinlin Hu, Pascal Fua, Mathieu Salzmann
Computer Vision Laboratory, EPFL, Lausanne, Switzerland.

NeurIPS 2019

Background

Eigendecomposition (ED) methods in deep networks:

- Standard ED with SVD (for symmetric positive semidefinite matrices) or QR decomposition
- Approximating it with the Power Iteration method

Both of them are numerically unstable, particularly when dealing with large matrices. While this can be mitigated by partitioning the data in small arbitrary groups, doing so has no theoretical basis and makes it impossible to exploit the power of ED to the full.

Analysis

Standard ED:

- Forward Pass: Perform ED using SVD or QR decomposition.
- Backpropagation: Use analytical gradients. Let λ_i denotes the i-th largest eigenvalue of M. The partial derivative of standard ED involves:

$$\widetilde{K}_{i,j} = \begin{cases} 1 \big/ (\lambda_i - \lambda_j) & if\ i \neq j \\ 0 & if\ i = j \end{cases}$$

Problem:

- ✓ when two eigenvalues are close: $\lambda_i - \lambda_j \to 0$, the partial derivatives become very large, causing ***arithmetic overflow***.

PI method:

- Forward Pass: Given random initial guesses for the eigenvectors, run a PI deflation procedure during the forward pass.
  - Given matrix $M = V\Sigma V^T$, PI computes leading eigenvector v iterative $\mathbf{v}^{(k)} = \frac{\mathbf{M}\mathbf{v}^{(k-1)}}{\|\mathbf{M}\mathbf{v}^{(k-1)}\|}$
- Backpropagation: Compute analytical gradients using the approximated eigenvectors and their intermediate values in the iteration process.

Problem:

- ✓ Inaccuracy: Its accuracy depends on the initial vectors and on the number of iterations in each step of the deflation procedure.
- ✓ Training Divergence: PI convergence rate depends geometrically on the ratio $|\lambda_2/\lambda_1|$ of the two largest eigenvalues. When the ratio is close to 1, the eigenvector may be ***inaccurate*** given the limited number of iterations.
- ✓ Round-off Error: The deflation procedure gradually removes the dominant eigenvectors and this will results in accumulated round-off error. The eigenvectors correspond to the small eigenvalues will be inaccurate.

Our Solution

Hybrid Strategy:

- Forward Pass: Use SVD. By relying on a divide-and-conquer strategy, SVD tends to be numerically more stable than QR decomposition.
- Backpropagation: Compute the gradients for backpropagation from the PI derivations, but using the SVD-computed vectors for initialization purposes.

Advantages:

- ✓ The resulting PI gradients converge to the analytical ED ones;
- ✓ Gradients are bounded by a factor depending on the number of PI iterations, thus preventing their explosion problem in practice.

SVD Gradients vs PI Gradients

SVD Gradients

$$\frac{\partial L}{\partial \mathbf{M}} = \sum_{i=2}^{n} \frac{1}{\lambda_1 - \lambda_i} \mathbf{v}_i \mathbf{v}_i^\top \frac{\partial L}{\partial \mathbf{v}_1} \mathbf{v}_1^\top \quad (1)$$

Reminder: Eq.(1) – Eq.(2)

$$\sum_{i=2}^{n} \frac{1}{\lambda_1 - \lambda_i} \left(\frac{\lambda_i}{\lambda_1}\right)^K \mathbf{v}_i \mathbf{v}_i^\top \frac{\partial L}{\partial \mathbf{v}_1} \mathbf{v}_1^\top \quad (3)$$

PI Gradients

$$\begin{aligned} \frac{\partial L}{\partial \mathbf{M}} &= \sum_{k=0}^{K-1} \frac{\left(\mathbf{I} - \mathbf{v}^{(k+1)}\mathbf{v}^{(k+1)\top}\right)}{\|\mathbf{M}\mathbf{v}^{(k)}\|} \frac{\partial L}{\partial \mathbf{v}^{(k+1)}} \mathbf{v}^{(k)\top} \\ &= \left(\frac{\left(\mathbf{I} - \mathbf{v}\mathbf{v}^\top\right)}{\|\mathbf{M}\mathbf{v}\|} + \frac{\mathbf{M}\left(\mathbf{I} - \mathbf{v}\mathbf{v}^\top\right)}{\|\mathbf{M}\mathbf{v}\|^2} + \cdots + \frac{\mathbf{M}^{K-1}\left(\mathbf{I} - \mathbf{v}\mathbf{v}^\top\right)}{\|\mathbf{M}\mathbf{v}\|^K} \right) \frac{\partial L}{\partial \mathbf{v}^{(K)}} \mathbf{v}^\top \\ &= \left(\sum_{i=2}^{n} \left(\frac{1}{\lambda_1} + \frac{1}{\lambda_1}\left(\frac{\lambda_i}{\lambda_1}\right)^1 + \cdots + \frac{1}{\lambda_1}\left(\frac{\lambda_i}{\lambda_1}\right)^{K-1} \right) \mathbf{v}_i \mathbf{v}_i^\top \right) \frac{\partial L}{\partial \mathbf{v}_1^{(K)}} \mathbf{v}_1^\top \end{aligned} \quad (2)$$

PI Gradients = geometric expansion of SVD gradients & PI gradients are ***bounded****.*

Let n denote matrix dimension, K denote the power iteration number.

Choosing a value of K = Choosing an **upper bound** for the gradient magnitudes.

$$\left\|\frac{\partial L}{\partial \mathbf{M}}\right\| \leq \left\| \sum_{i=2}^{n} \left(\frac{1}{\lambda_1} \cdots + \frac{1}{\lambda_1} \right) \mathbf{v}_i \mathbf{v}_i^\top \right\| \left\|\frac{\partial L}{\partial \mathbf{v}_1}\right\| \|\mathbf{v}_1^\top\| \leq \sum_{i=2}^{n} \left\| \frac{K}{\lambda_1} \mathbf{v}_i \mathbf{v}_i^\top \right\| \left\|\frac{\partial L}{\partial \mathbf{v}_1}\right\| \|\mathbf{v}_1^\top\| \leq \frac{nK}{\lambda_1} \left\|\frac{\partial L}{\partial \mathbf{v}_1}\right\| \rightarrow \left\|\frac{\partial L}{\partial (\mathbf{M} + \epsilon I)}\right\| \leq \frac{nK}{\epsilon} \left\|\frac{\partial L}{\partial \mathbf{v}_1}\right\|$$

Ratio between Reminder (3) and Original value (1) is controlled by $\left(\lambda_i/\lambda_1\right)^K$

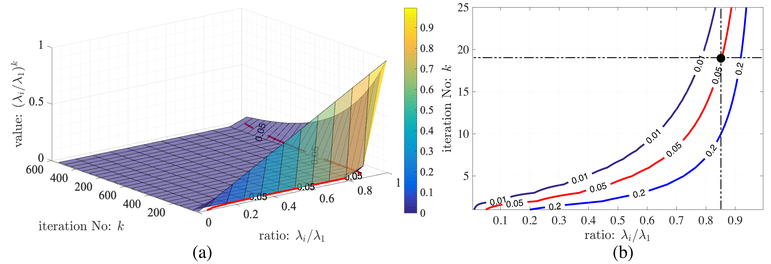


Figure: (a) shows how value of $(\lambda_i/\lambda_1)^k$ changes w.r.t. the eigenvalue ratio λ_i/λ_1 and iteration number k.
(b) shows the contour of curved surface in (a).

Experiment

ZCA Whitening: $Z = WX = U\Lambda^{-1/2}U^TX$;
U, Λ are the eigenvectors and eigenvalues of M, M is the covariance matrix of X.
ZCA is a transformation which makes the covariance matrix Σ to be the identity matrix.
Hence it decorrelates features.

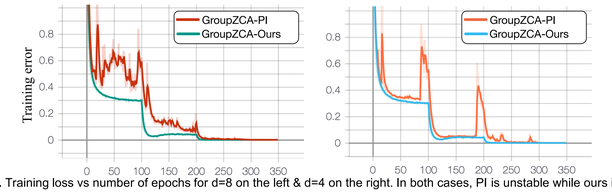


Figure 1. Training loss vs number of epochs for d=8 on the left & d=4 on the right. In both cases, PI is unstable while ours is stable.

Methods	Error	Matrix Dimension				
		$d=4$	$d=8$	$d=16$	$d=32$	$d=64$
SVD	Min, Mean	4.59, **4.54 ± 0.08**	-	-	-	-
	Suc. Rate	46.7%	0%	0%	0%	0%
PI	Min, Mean	4.44, 4.99±0.51	6.28,-	-	-	-
	Suc. Rate	100%	6.7%	0%	0%	0%
Ours	Min, Mean	4.59, 4.71±0.11	4.43, 4.62±0.18	**4.40**, 4.63±0.14	4.46, 4.64±0.15	4.44, **4.59 ± 0.09**
	Suc. Rate	100%	100%	100%	100%	100%

Table 1. Errors and success rates using ResNet 18 with SVD, PI, and our method on CIFAR10. d is the size of feature groups.

PCA Denoising: $Z = \widetilde{W}X = \widetilde{U_k}\widetilde{U_k}^TX$;
$\widetilde{U_k}$ is the top k eigenvectors of the covariance matrix of X.

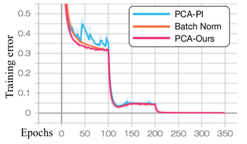
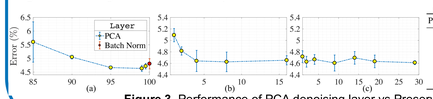


Figure 2. Training loss vs number of epochs.

Methods	Error	Value
BN	Min	4.66
	Mean	4.81±0.19
PCA(PI)	Min	5.05
	Mean	5.35±0.25
PCA(SVD)	Min	NaN
	Mean	NaN
PCA(Ours)	Min	**4.58**
	Mean	**4.63±0.11**

Table 2. Final error rate.

- In Fig.2, the loss for our method and BN decrease smoothly, meaning the training process is very stable. By contrast, the PI training curve denotes far more instability and lower performance.
- In Table. 2, the NaN values associated to SVD denotes the fact that using the analytical SVD derivatives failed all five times we tried.



(a) Preserved Information	
Percentage (%)	Error (%)
85	5.60 ± 0.74
90	5.05 ± 0.07
95	4.67 ± 0.06
99	**4.63 ± 0.11**
99.5	4.72 ± 0.10
100	4.81 ± 0.19

(b) Eigenvector No.	
Eig-vector No.	Error (%)
2	5.09 ± 0.12
4	4.82 ± 0.07
8	4.64 ± 0.18
16	**4.63 ± 0.17**
32	4.65 ± 0.08

(c) Power Iteration No.	
Power Iteration No.	Error (%)
1	4.72 ± 0.10
2	4.63 ± 0.09
4	4.67 ± 0.09
9	4.61 ± 0.14
14	4.69 ± 0.11
19	4.63 ± 0.11
29	**4.61 ± 0.03**

Figure 3. Performance of PCA denoising layer vs Preserved information, Eigenvector No, and PI Number.

Conclusion

We have introduced a numerically stable differentiable ED method that relies on the SVD during the forward pass and on PI to compute the gradients during the backward pass. Both the theory and the experimental results confirm the increased stability that our method brings compared with standard SVD or PI alone.