Large-Margin Softmax Loss for Convolutional Neural Networks

Weiyang Liu[1*], Yandong Wen[2*], Zhiding Yu[3] and Meng Yang[4]

[1]Peking University [2]South China University of Technology

[3]Carnegie Mellon University [4]Shenzhen University [*]equal contribution

Introduction

Cross-entropy loss together with softmax is arguably one of the most common used supervision components in convolutional neutral networks (CNNs). Despite its simplicity, popularity and excellent performance, the component does not explicitly encourage discriminative learning of features. In this paper, we propose a generalized large-margin softmax (L-Softmax) loss which explicitly encourages intra-class compactness and inter-class separability between learned features. Moreover, L-Softmax not only can adjust the desired margin but also can avoid overfitting. We also show that the L-Softmax loss can be optimized by typical stochastic gradient descent. Extensive experiments on four benchmark datasets demonstrate that the deeply-learned features with L-Softmax loss become more discriminative, hence significantly boosting the performance on a variety of visual classification and verification tasks.

From Softmax Loss to Large-Margin Softmax Loss

Standard softmax loss can be written as

$$L = \frac{1}{N}\sum_i L_i = \frac{1}{N}\sum_i -\log\left(\frac{e^{f_{y_i}}}{\sum_j e^{f_j}}\right)$$

Using the transformation of inner products, the softmax loss is reformulated as

$$L_i = -\log\left(\frac{e^{\|\boldsymbol{W}_{y_i}\|\|\boldsymbol{x}_i\|\cos(\theta_{y_i})}}{\sum_j e^{\|\boldsymbol{W}_j\|\|\boldsymbol{x}_i\|\cos(\theta_j)}}\right)$$

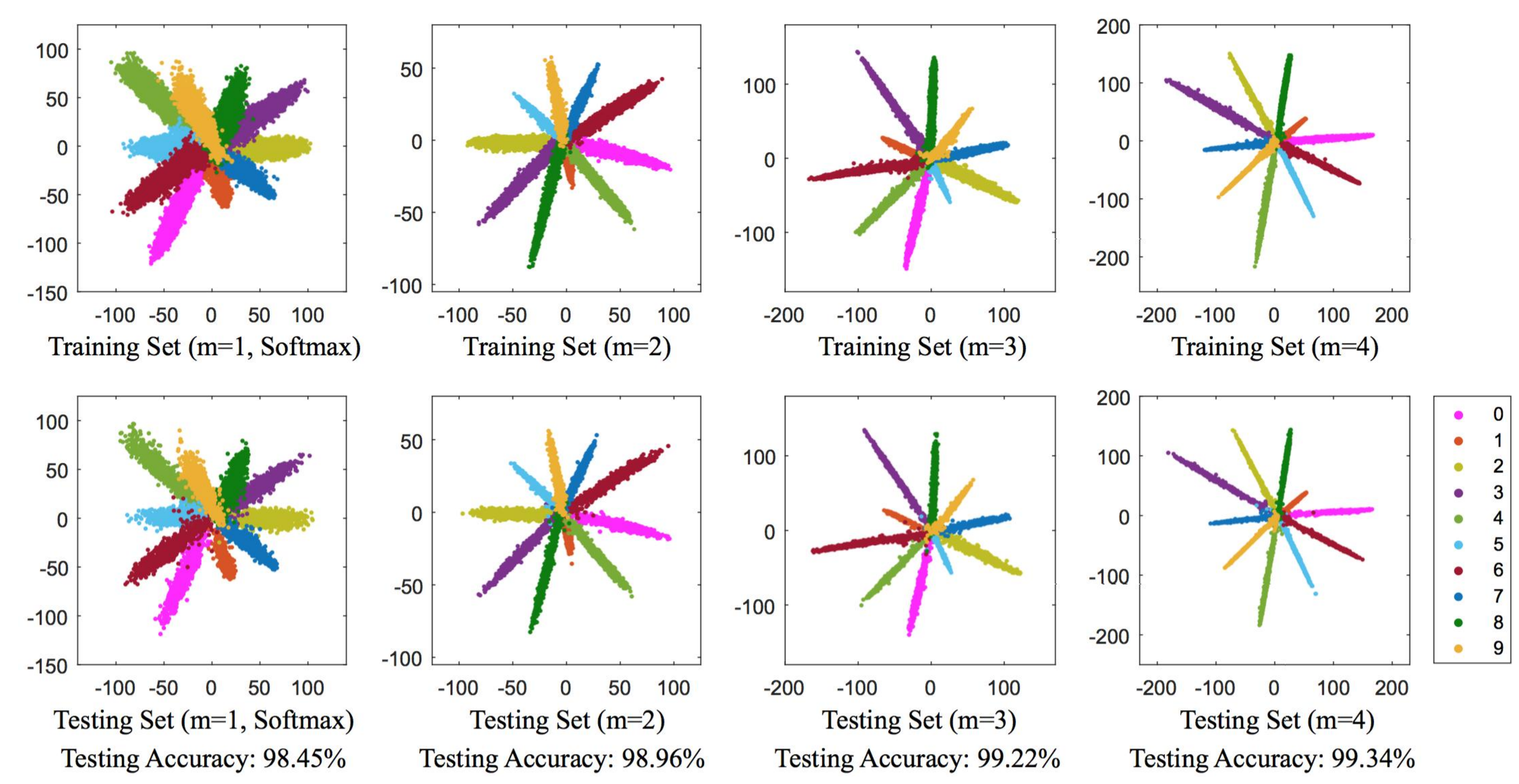
The large-margin softmax loss is formulated as

$$L_i = -\log\left(\frac{e^{\|\boldsymbol{W}_{y_i}\|\|\boldsymbol{x}_i\|\psi(\theta_{y_i})}}{e^{\|\boldsymbol{W}_{y_i}\|\|\boldsymbol{x}_i\|\psi(\theta_{y_i})} + \sum_{j\neq y_i} e^{\|\boldsymbol{W}_j\|\|\boldsymbol{x}_i\|\cos(\theta_j)}}\right)$$

where $\psi(\theta) = (-1)^k \cos(m\theta) - 2k, \quad \theta \in [\frac{k\pi}{m}, \frac{(k+1)\pi}{m}]$

Intuition & Geometric Interpretation

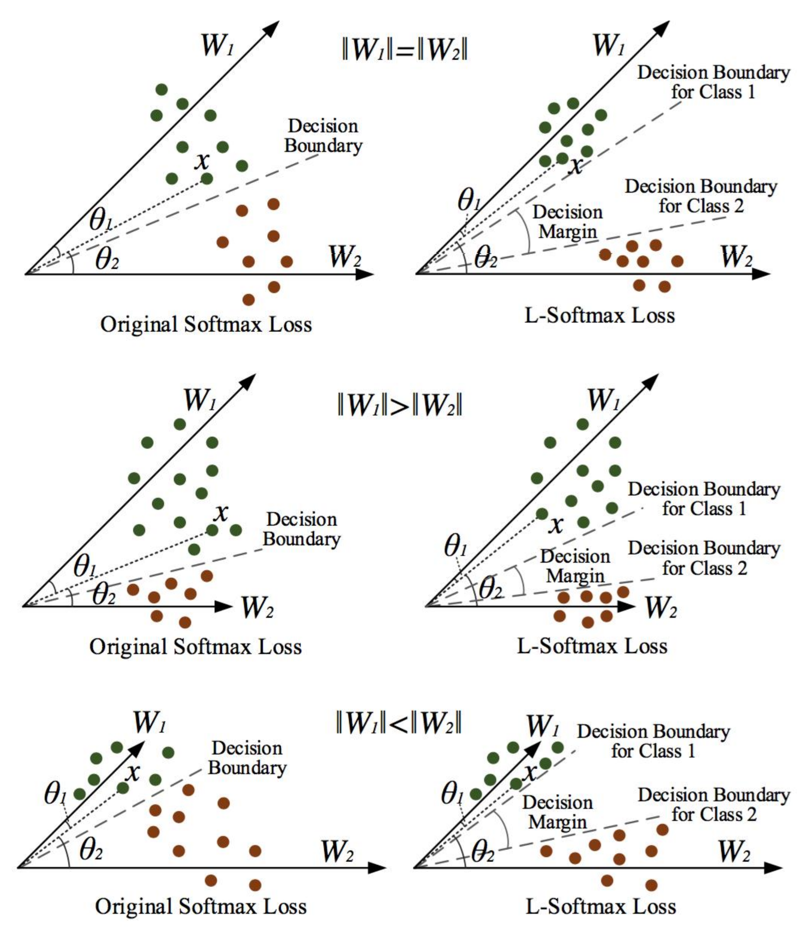
The purpose of L-Softmax loss is to learn discriminative features with large angular margin. We train the CNN with L-Softmax loss on MNIST dataset. The deeply-learned features are visualized in the following figure.



One can observe that the features learned via L-Softmax loss are indeed more discriminative than those learned via standard softmax loss.

The geometric interpretation is given on the right. The L-Softmax can produce an angular decision margin between different classes, because it requires more rigorous classification criteria compared to the standard softmax loss.

- ➢ The parameter m controls the desired decision margin.
- ➢ The L-Softmax can be easily optimized using SGD.
- ➢ It can be used in tandem with other regularization methods.



Experiments & Results

We perform extensive experiments on visual classification and face verification task, achieving state-of-the-art results on MNIST, CIFAR10, CIFAR100 and LFW public datasets.

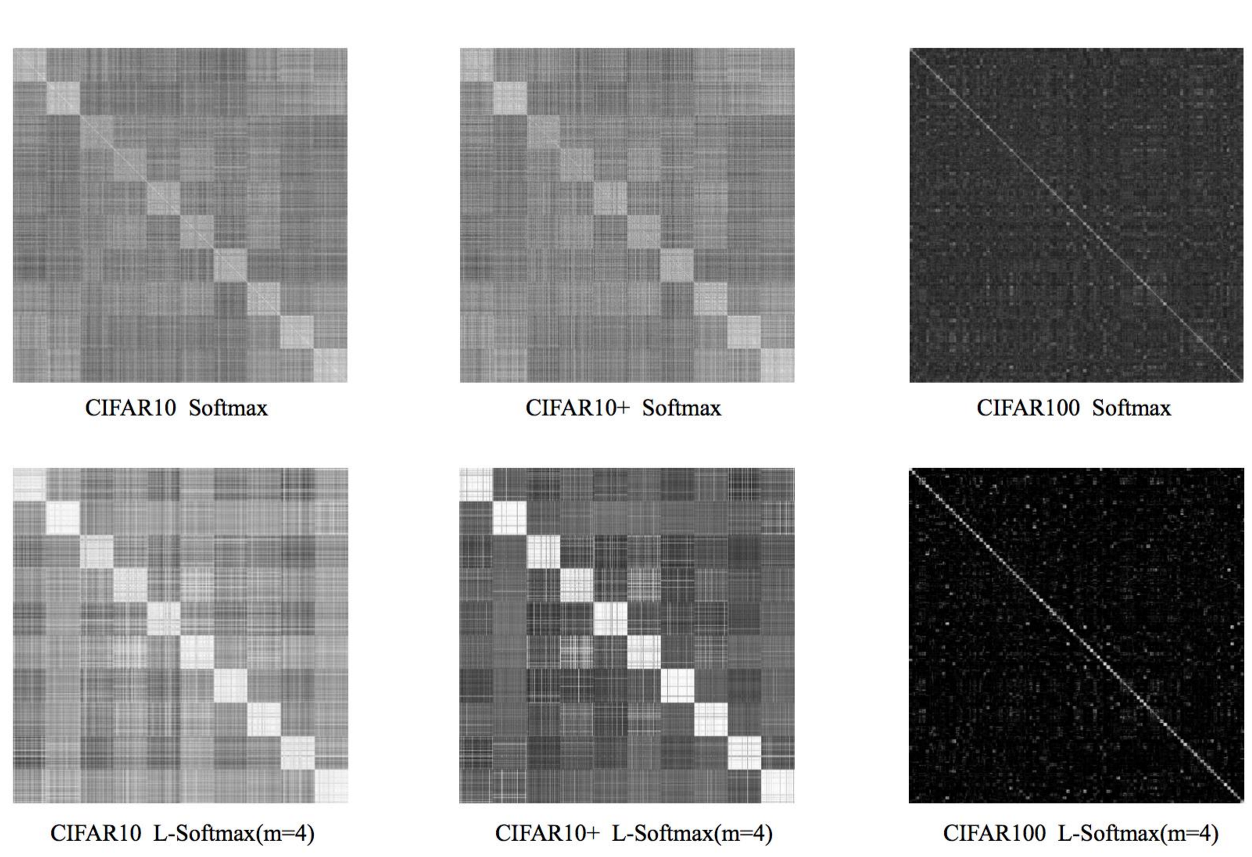
- ➢ On all these datasets, we have shown that the classification accuracy will be improved with larger m, namely when the desired decision margin is set to be larger.
- ➢ The confusion matrix on CIFAT10, CIFAR10+ and CIFAR100 validate the discriminativeness of the deeply-learned features via our proposed L-Softmax loss.

Method	Error Rate
CNN (Jarrett et al., 2009)	0.53
DropConnect (Wan et al., 2013)	0.57
FitNet (Romero et al., 2015)	0.51
NiN (Lin et al., 2014)	0.47
Maxout (Goodfellow et al., 2013)	0.45
DSN (Lee et al., 2015)	0.39
R-CNN (Liang & Hu, 2015)	**0.31**
GenPool (Lee et al., 2016)	**0.31**
Hinge Loss	0.47
Softmax	0.40
L-Softmax (m=2)	0.32
L-Softmax (m=3)	**0.31**
L-Softmax (m=4)	**0.31**

Accuracy (%) on MNIST

Method	CIFAR10	CIFAR10+
DropConnect (Wan et al., 2013)	9.41	9.32
FitNet (Romero et al., 2015)	N/A	8.39
NiN + LA units (Lin et al., 2014)	10.47	8.81
Maxout (Goodfellow et al., 2013)	11.68	9.38
DSN (Lee et al., 2015)	9.69	7.97
All-CNN (Springenberg et al., 2015)	9.08	7.25
R-CNN (Liang & Hu, 2015)	8.69	7.09
ResNet (He et al., 2015a)	N/A	6.43
GenPool (Lee et al., 2016)	7.62	6.05
Hinge Loss	9.91	6.96
Softmax	9.05	6.50
L-Softmax (m=2)	7.73	6.01
L-Softmax (m=3)	7.66	5.94
L-Softmax (m=4)	**7.58**	**5.92**

Accuracy (%) on CIFAR10 & CIFAR10+



Confusion matrix on CIFAR10, CIFAR10+, CIFAR100

Method	Error Rate
FitNet (Romero et al., 2015)	35.04
NiN (Lin et al., 2014)	35.68
Maxout (Goodfellow et al., 2013)	38.57
DSN (Lee et al., 2015)	34.57
dasNet (Stollenga et al., 2014)	33.78
All-CNN (Springenberg et al., 2015)	33.71
R-CNN (Liang & Hu, 2015)	31.75
GenPool (Lee et al., 2016)	32.37
Hinge Loss	32.90
Softmax	32.74
L-Softmax (m=2)	29.95
L-Softmax (m=3)	29.87
L-Softmax (m=4)	**29.53**

Accuracy (%) on CIFAR100

Method	Outside Data	Accuracy
FaceNet (Schroff et al., 2015)	200M*	**99.65**
Deep FR (Parkhi et al., 2015)	2.6M	98.95
DeepID2+ (Sun et al., 2015)	300K*	98.70
(Yi et al., 2014)	WebFace	97.73
(Ding & Tao, 2015)	WebFace	98.43
Softmax	WebFace	96.53
Softmax + Contrastive	WebFace	97.31
L-Softmax (m=2)	WebFace	97.81
L-Softmax (m=3)	WebFace	98.27
L-Softmax (m=4)	WebFace	**98.71**

Verification accuracy(%) on LFW