CosFace

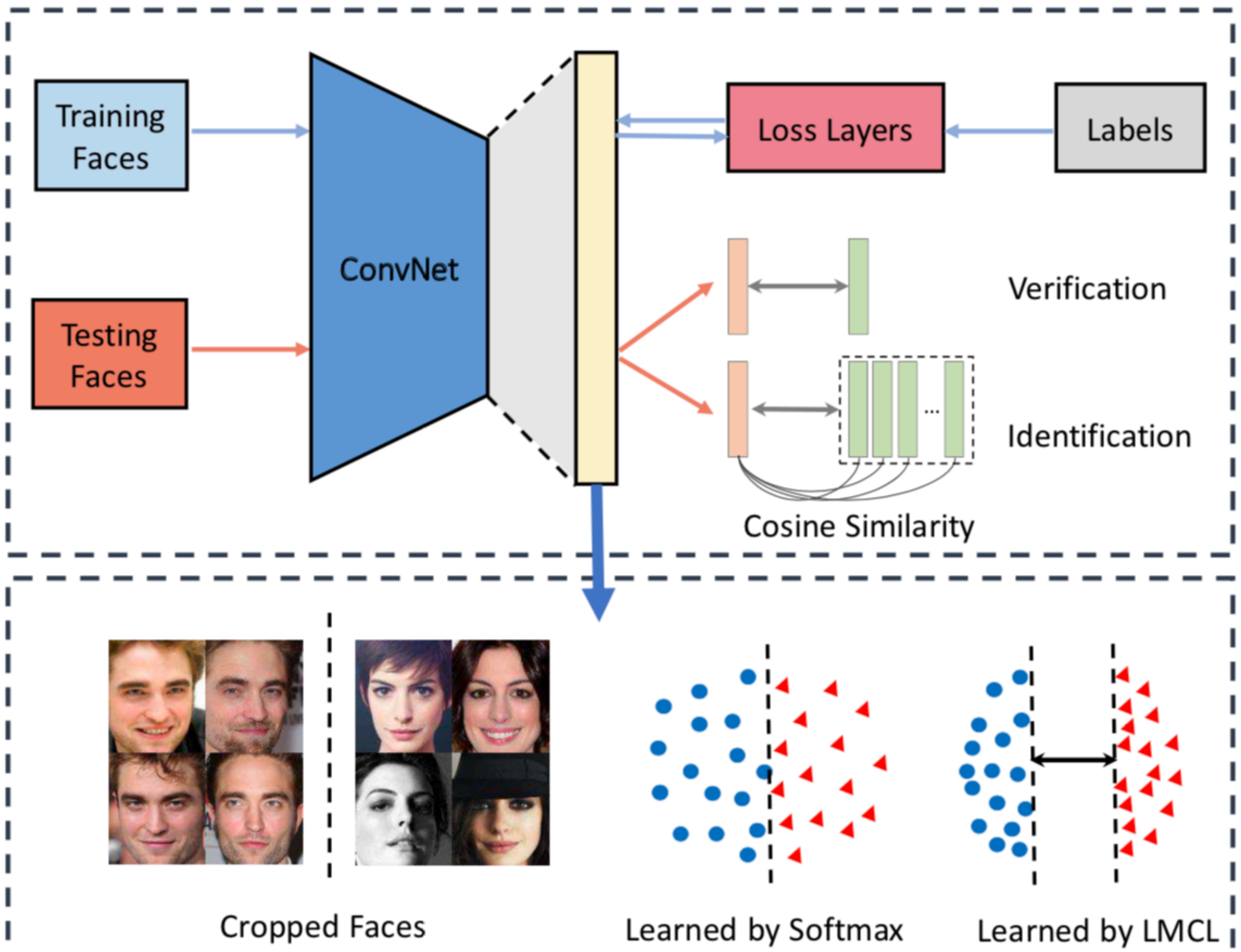
Cho Sung Man

Index

- Introduction
- LMCL(Large Margin Cosine Loss)
- Experiments

Introduction

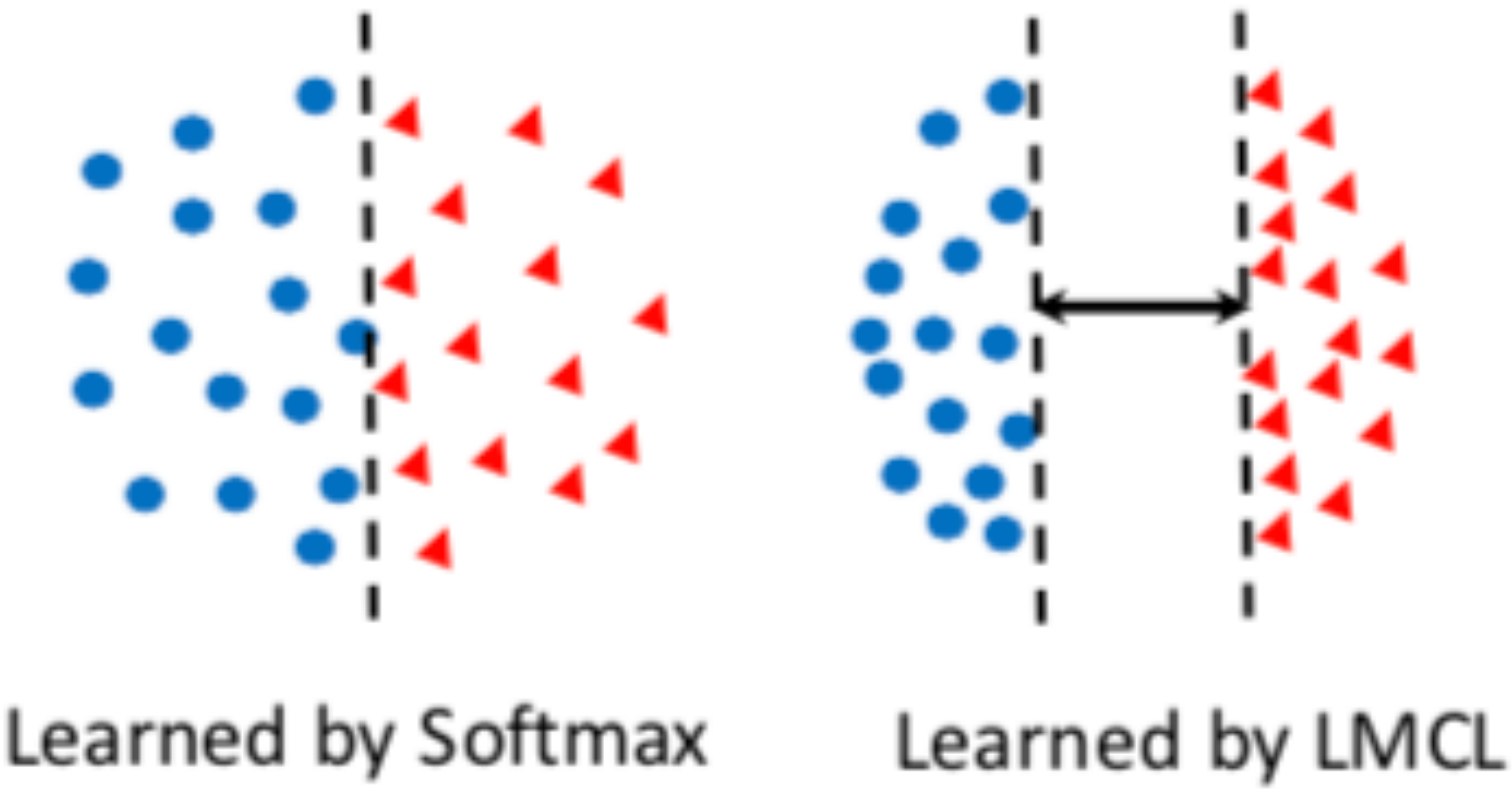
In this presentation,



- **Traditional soft-max in the face recognition task**
- **Several loss functions**
- **LMCL (Large Margin Cosine Loss)**

Traditional Problem

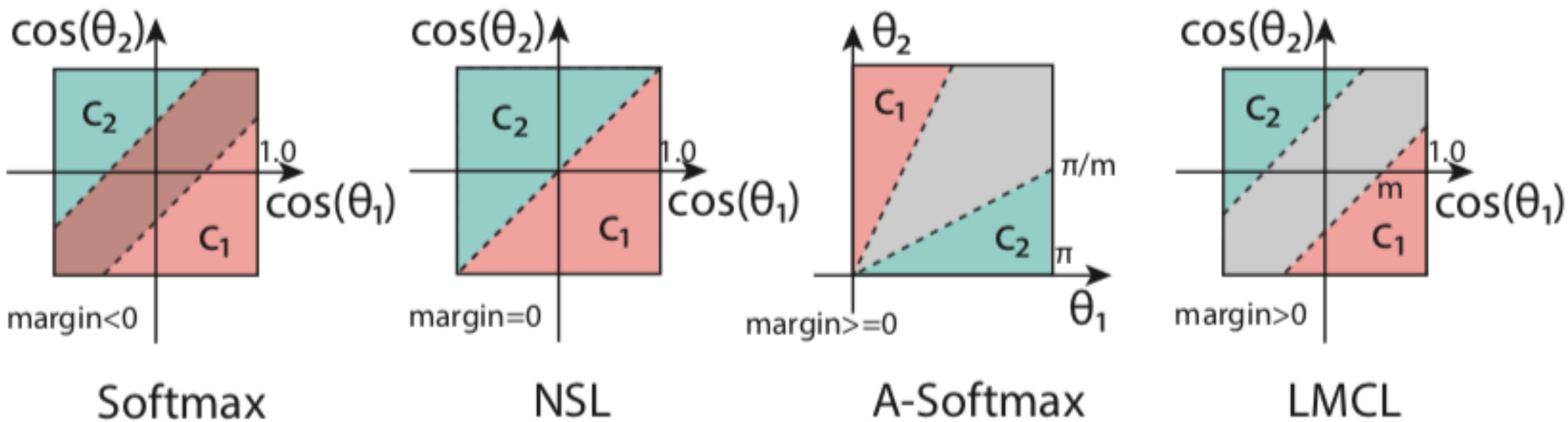
In face recognition task,



Maximizing inter-class variance
Minimizing intra-class variance

Other Loss Functions

Softmax, NSL, A-Softmax



Compared to the Euclidean margin, the angular margin is preferred because the cosine of the angle has intrinsic consistency with softmax.

LMCL

LMCL

The softmax loss separates features from different classes by **maximizing the posterior probability of the ground-truth class.**

Posterior probability

$$L_s = \frac{1}{N}\sum_{i=1}^{N} -\log p_i = \frac{1}{N}\sum_{i=1}^{N} -\log \frac{e^{f_{y_i}}}{\sum_{j=1}^{C} e^{f_j}},$$

Training Sample

$$f_j = W_j^T x = \|W_j\| \|x\| \cos\theta_j,$$

Both norm and angle of vectors contribute to the posterior prob.

LMCL

$$L_s = \frac{1}{N}\sum_{i=1}^{N} -\log p_i = \frac{1}{N}\sum_{i=1}^{N} -\log \frac{e^{f_{y_i}}}{\sum_{j=1}^{C} e^{f_j}},$$

To develop effective feature learning, L2 normalization.

$$\|W_j\| = 1$$

norm of feature vector x is not contributing to the scoring function.

$$\|x\| = s$$

$$L_{ns} = \frac{1}{N}\sum_{i} -\log \frac{e^{s\cos(\theta_{y_i,i})}}{\sum_j e^{s\cos(\theta_{j,i})}}.$$

LMCL

$$L_s = \frac{1}{N}\sum_{i=1}^{N} -\log p_i = \frac{1}{N}\sum_{i=1}^{N} -\log \frac{e^{f_{y_i}}}{\sum_{j=1}^{C} e^{f_j}},$$

To develop effective feature learning, L2 normalization.

$$\|W_j\| = 1$$

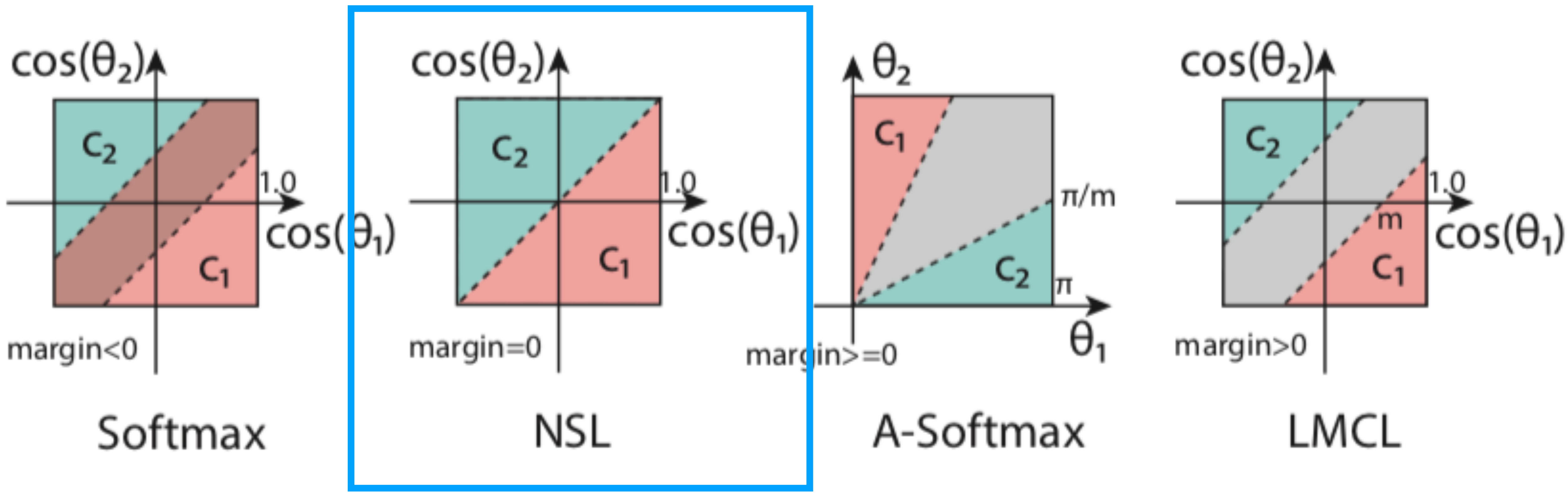
Normalized version of Softmax Loss (NSL)

norm of feature vector x is not contributing to the scoring function.

$$\|x\| = s$$

$$L_{ns} = \frac{1}{N}\sum_{i} -\log \frac{e^{s\cos(\theta_{y_i,i})}}{\sum_j e^{s\cos(\theta_{j,i})}}.$$

NSL



NSL only emphasizes correct classification

LMCL

Considering a scenario of binary-classes, NSL forces (for C1)

$$\cos(\theta_1) \; > \; \cos(\theta_2)$$

To develop a large margin classifier, we further require

$$\cos(\theta_1) - m > \cos(\theta_2)$$

Formally, we define the Large Margin Cosine Loss,

$$L_{lmc} = \frac{1}{N} \sum_i -\log \frac{e^{s(\cos(\theta_{y_i,i}) - m)}}{e^{s(\cos(\theta_{y_i,i}) - m)} + \sum_{j \neq y_i} e^{s \cos(\theta_{j,i})}},$$

subject to,

$$W = \frac{W^*}{\|W^*\|}, \quad x = \frac{x^*}{\|x^*\|}, \quad cos(\theta_j, i) = {W_j}^T x_i,$$

Comparisons

Softmax : $$\|W_1\|\cos(\theta_1) = \|W_2\|\cos(\theta_2).$$ overlap !

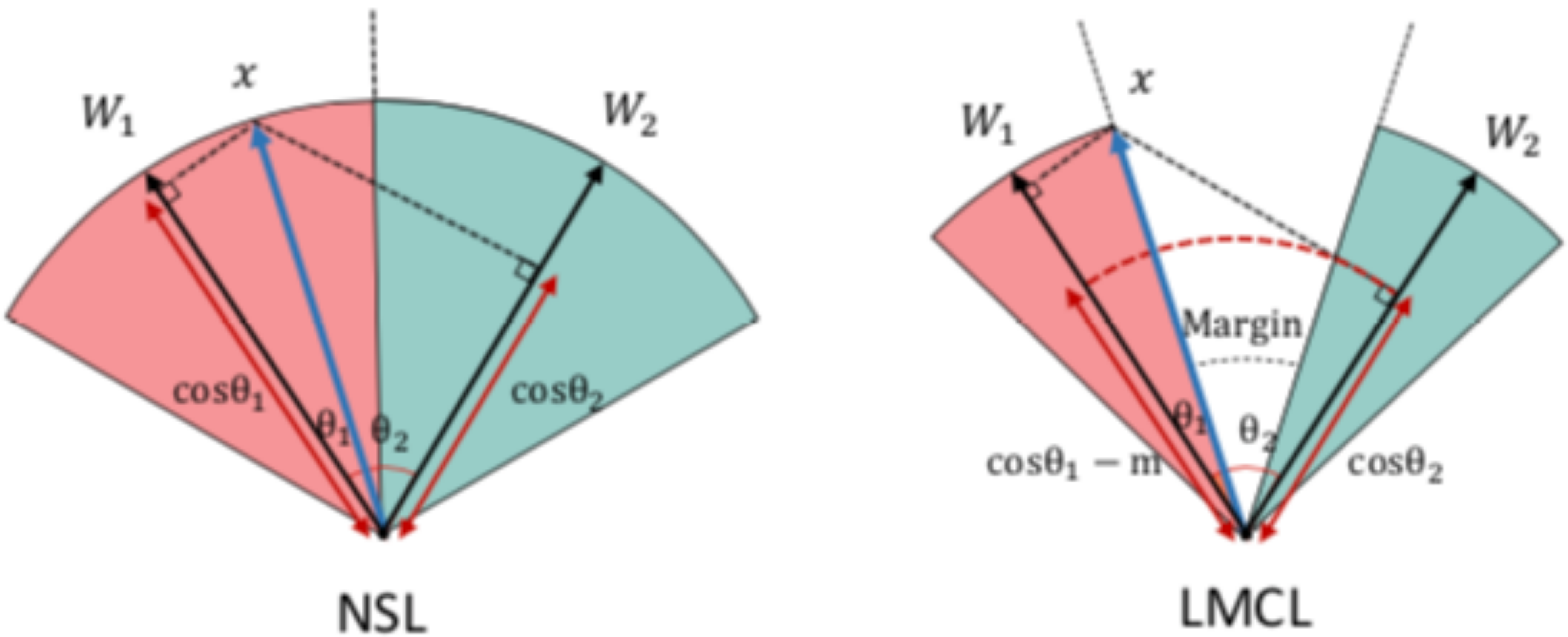
NSL : $$\cos(\theta_1) = \cos(\theta_2).$$ Weight Normalization

A-Softmax : $$C_1 : \cos(m\theta_1) \geq \cos(\theta_2),$$ $$C_2 : \cos(m\theta_2) \geq \cos(\theta_1).$$ if theta = 0 ?

LMCL : $$C_1 : \cos(\theta_1) \geq \cos(\theta_2) + m,$$ $$C_2 : \cos(\theta_2) \geq \cos(\theta_1) + m.$$

Normalization on Features

LMCL simultaneously normalizes both weight vectors and feature vectors.



learning only depends on cosine values to develop the discriminative power.

Why Normalization?

$$L_{lmc} = \frac{1}{N}\sum_i -\log \frac{e^{s(\cos(\theta_{y_i,i})-m)}}{e^{s(\cos(\theta_{y_i,i})-m)} + \sum_{j\neq y_i} e^{s\cos(\theta_{j,i})}},$$

Do NOT change ! for optimization

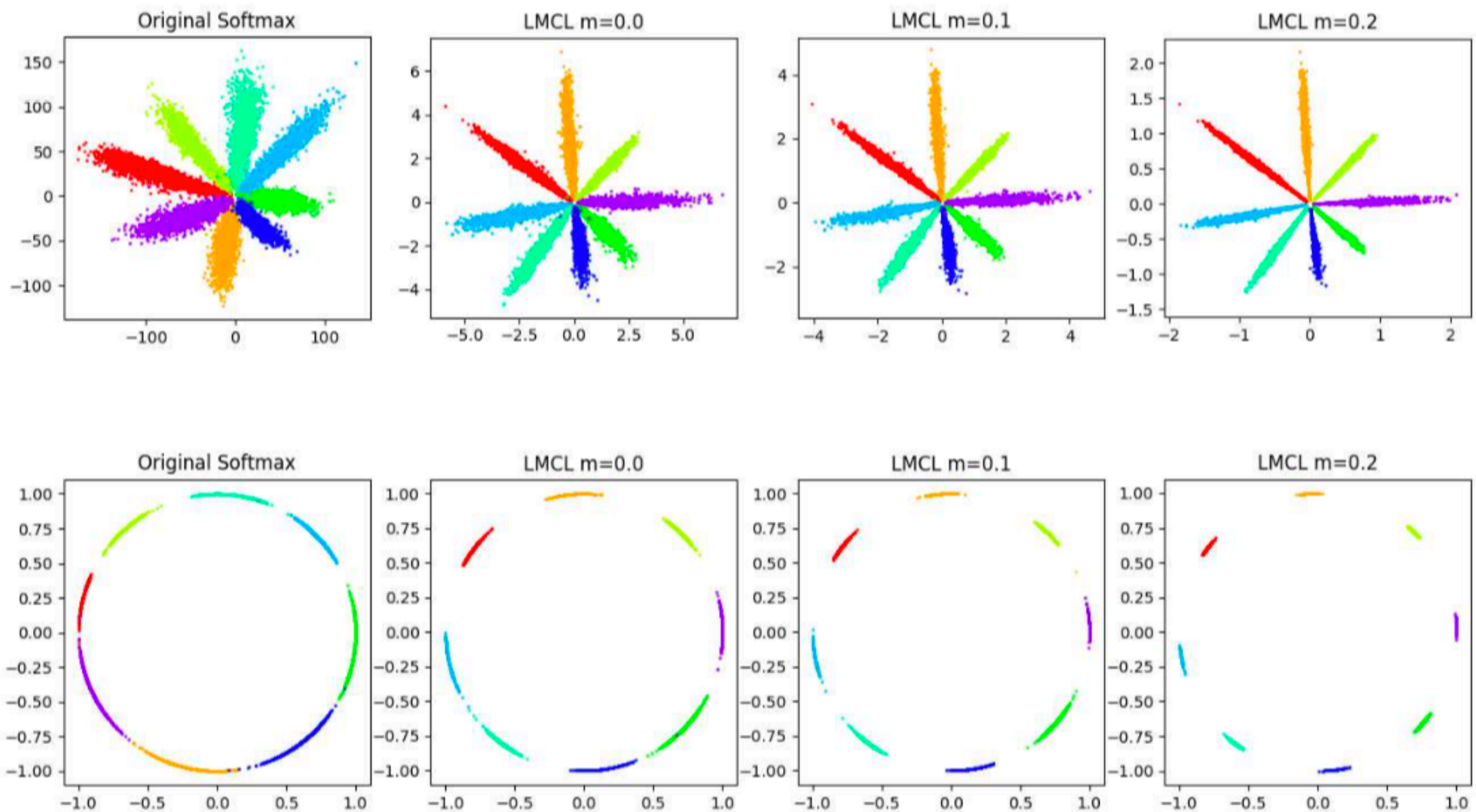
$$\boxed{\|x\|}(\cos(\theta_i) - m) > \boxed{\|x\|}\cos(\theta_j).$$

scaling parameter lower bound

$$s \geq \frac{C-1}{C}\log\frac{(C-1)P_W}{1-P_W}.$$

expected minimum posterior probability

2D features

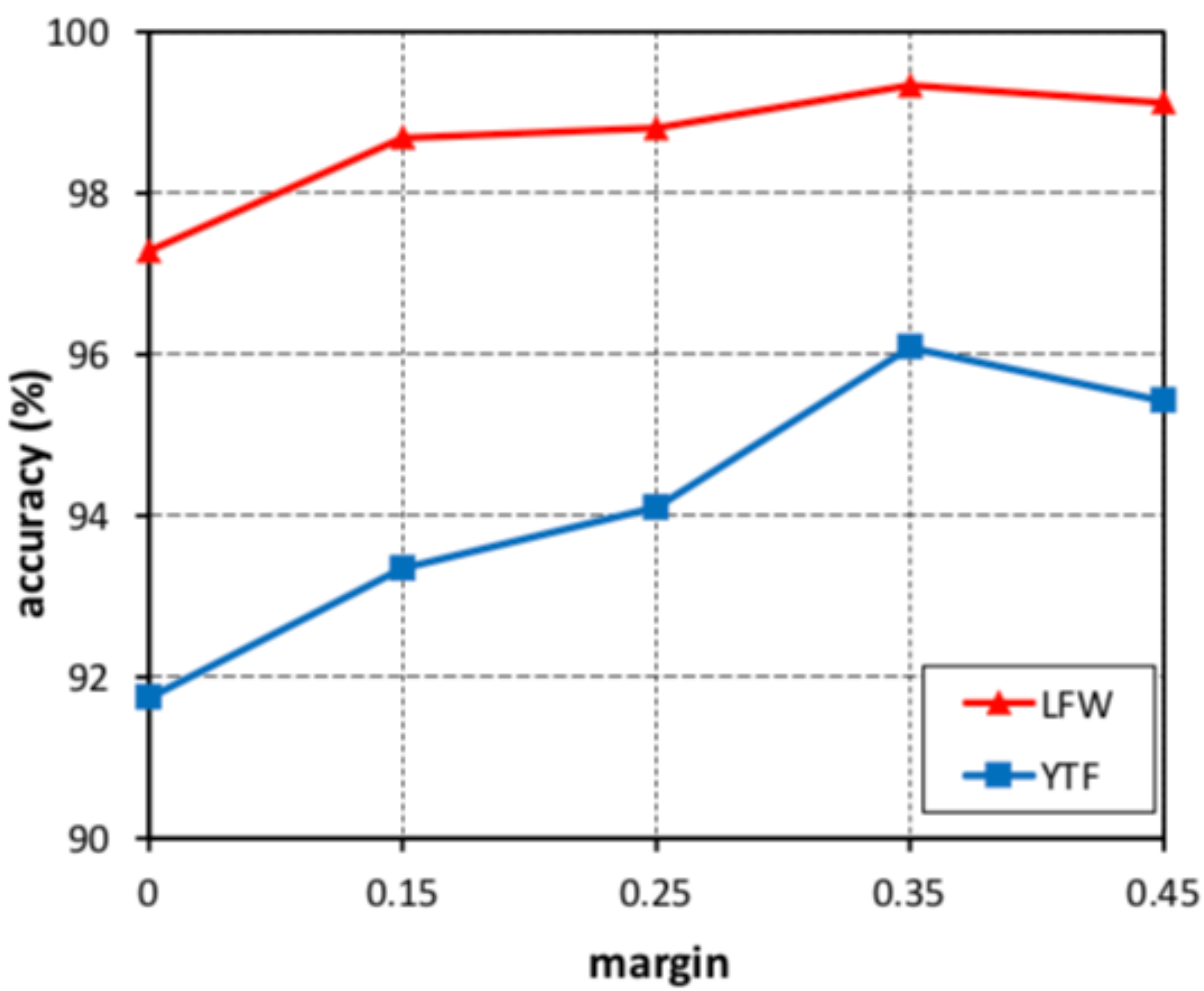


Original Softmax
LMCL m=0.0
LMCL m=0.1
LMCL m=0.2
Original Softmax
LMCL m=0.0
LMCL m=0.1
LMCL m=0.2

Experiments

Experiments

Accuracy of CosFace with different margin parameters on LFW and YTF



Experiments

Effect of feature normalization

Normalization	LFW	YTF	MF1 Rank 1	MF1 Veri.
No	99.10	93.1	75.10	88.65
Yes	99.33	96.1	77.11	89.88

with and without feature normalization

Benchmarks

Method	Protocol	MF1 Rank1	MF1 Veri.
SIAT_MMLAB[42]	Small	65.23	76.72
DeepSense - Small	Small	70.98	82.85
SphereFace - Small[23]	Small	75.76	90.04
Beijing FaceAll V2	Small	76.66	77.60
GRCCV	Small	77.67	74.88
FUDAN-CS_SDS[41]	Small	77.98	79.19
CosFace(Single-patch)	Small	77.11	89.88
CosFace(3-patch ensemble)	Small	**79.54**	**92.22**
Beijing FaceAll_Norm_1600	Large	64.80	67.11
Google - FaceNet v8[29]	Large	70.49	86.47
NTechLAB - facenx_large	Large	73.30	85.08
SIATMMLAB TencentVision	Large	74.20	87.27
DeepSense V2	Large	81.29	95.99
YouTu Lab	Large	83.29	91.34
Vocord - deepVo V3	Large	**91.76**	94.96
CosFace(Single-patch)	Large	82.72	96.65
CosFace(3-patch ensemble)	Large	84.26	**97.96**

Benchmarks

Method	Training Data	#Models	LFW	YTF
Deep Face[35]	4M	3	97.35	91.4
FaceNet[29]	200M	1	99.63	95.1
DeepFR [27]	2.6M	1	98.95	97.3
DeepID2+[33]	300K	25	99.47	93.2
Center Face[42]	0.7M	1	99.28	94.9
Baidu[21]	1.3M	1	99.13	-
SphereFace[23]	0.49M	1	99.42	95.0
CosFace	5M	1	**99.73**	**97.6**