One-Shot Learning

Alireza Akhavan Pour

AkhavanPour.ir
CLASS.VISION

One-Shot Learning

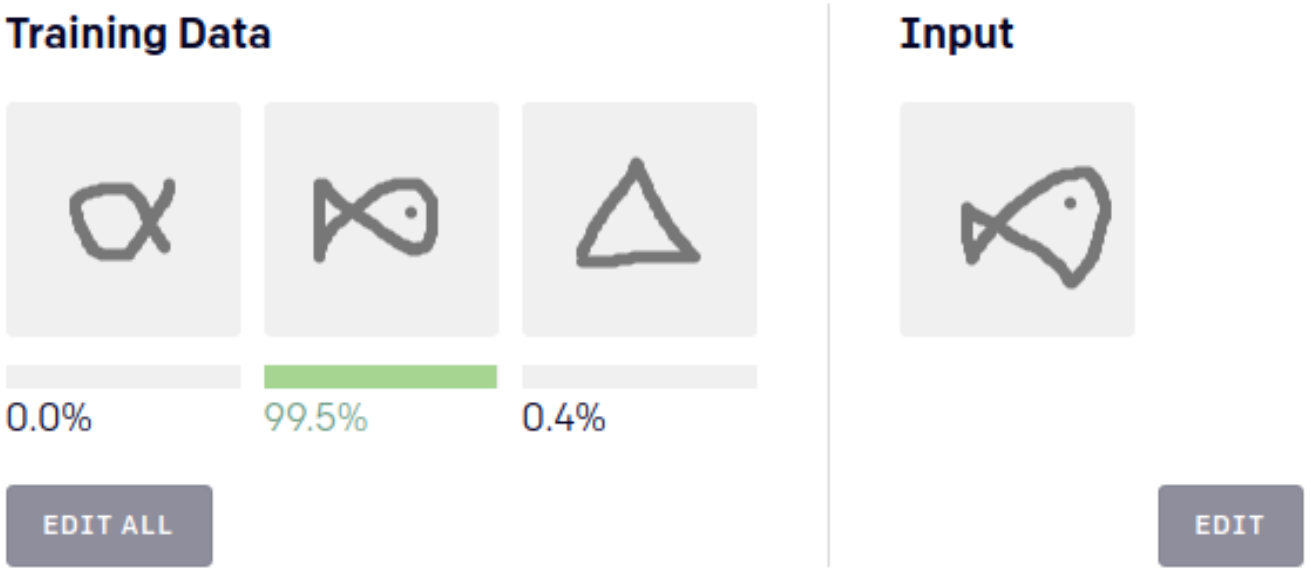
- One-shot learning is a classification task where **one**, or a **few, examples** are used to classify many new examples in the future.

The **Omniglot** dataset:

- 1,623 characters
- 50 different alphabets
- 20 examples for each character

https://github.com/brendenlake/omniglot/

meta-learning problem for few-shot classification



https://openai.com/blog/reptile/

It is hard!

- **Humans** learn new concepts with **very little supervision** – e.g. a child can generalize the concept of “giraffe” from a single picture in a book – yet our best deep learning systems need hundreds or thousands of examples.

— Matching Networks for One Shot Learning, 2017.

http://papers.nips.cc/paper/6385-matching-networks-for-one-shot-learning

One-shot learning is related to but different from zero-shot learning!

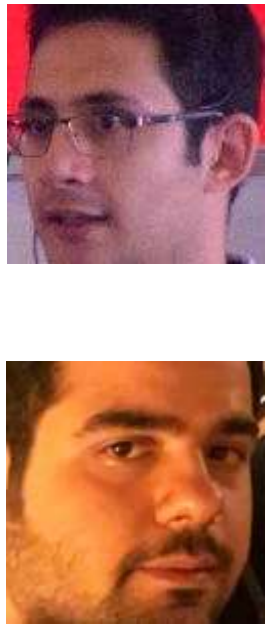
- This should be distinguished from zero-shot learning, in which the model cannot look at any examples from the target classes.

— Siamese Neural Networks for One-shot Image Recognition, 2015.

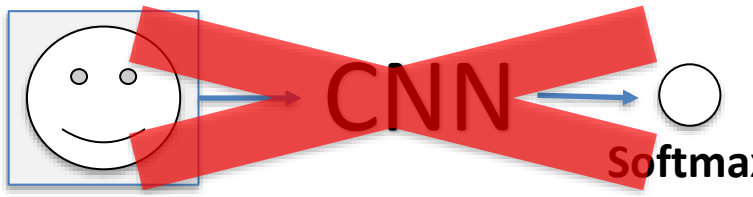
https://www.cs.cmu.edu/~rsalakhu/papers/oneshot1.pdf

One-shot learning

face recognition is a common example of one-shot learning



- آموزش از روی یک نمونه برای بازشناختن مجدد فرد

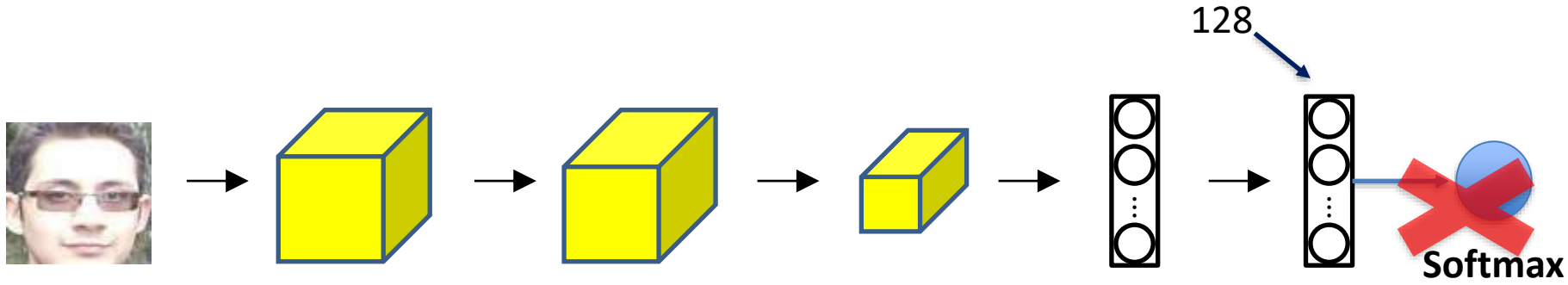


- مشکلات
  - تصاویر بسیار کم.
  - اگر فرد جدید اضافه شود؟

Siamese Network

Siamese Network for One-Shot Learning

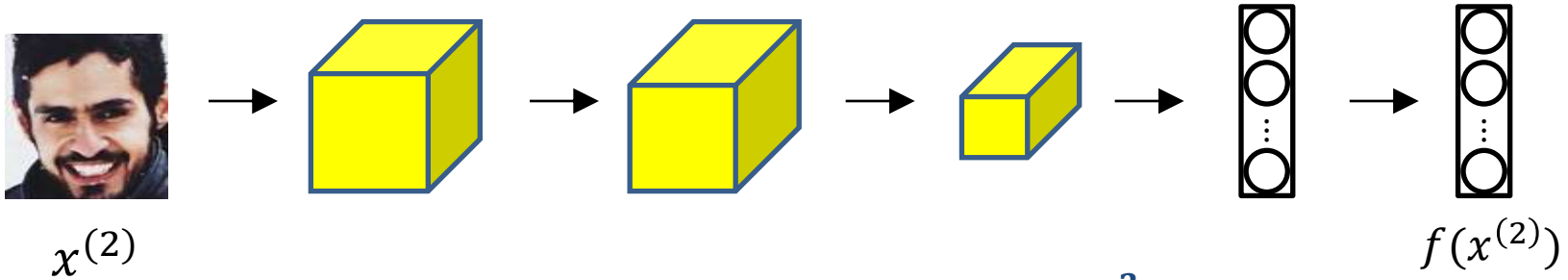
Siamese network



$x^{(1)}$

$f(x^{(1)})$

$f(x^{(1)})$: کدگذاری یا encoding تصویر $x^{(1)}$

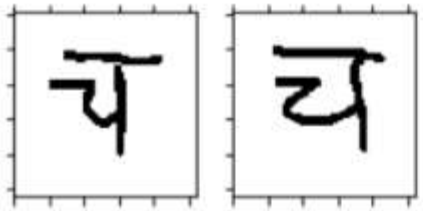
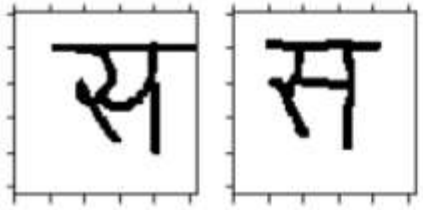


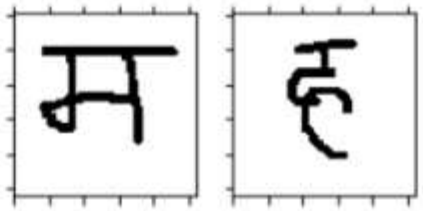
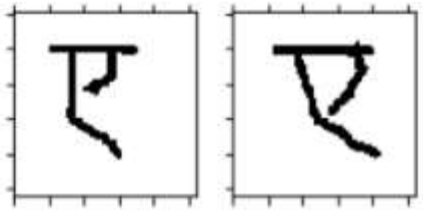
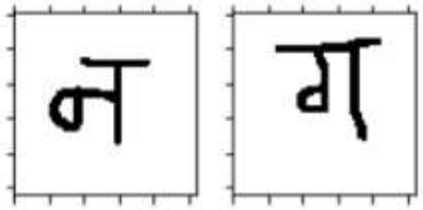
$x^{(2)}$

$f(x^{(2)})$

$$d(x^{(1)}, x^{(2)}) = \|f(x^{(2)}) - f(x^{(1)})\|_2^2$$

[Taigman et. al., 2014. DeepFace closing the gap to human level performance]

Xi		Labels (Yi)
		1
		0
		1

Xi		Labels (Yi)
		0
		1
		0

Let’s code…

- **1-OneShot(siamese).ipynb**

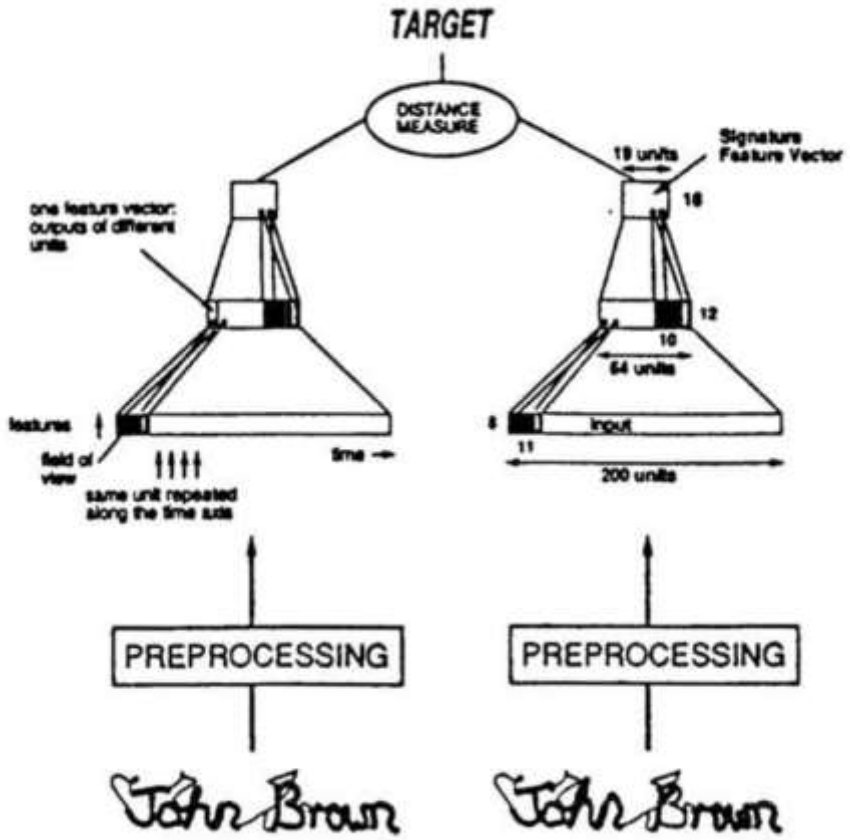
  https://colab.research.google.com/github/Alireza-Akhavan/deep-face-recognition/blob/master/aaiss2020/1-OneShot(siamese).ipynb

Signature Verification using a “Siamese”

- Verification consists of comparing an extracted feature vector with a stored feature vector for the signer. Signatures closer to this stored representation than a chosen threshold are accepted, all other signatures are rejected as forgeries.

- — Signature Verification using a “Siamese” Time Delay Neural Network, 1993.

http://papers.nips.cc/paper/769-signature-verification-using-a-siamese-time-delay-neural-network.pdf



Siamese Neural Networks for One-Shot Image Recognition

- Siamese networks were used more recently, where deep convolutional neural networks were used in parallel image inputs in a **2015** paper by **Gregory Koch**, et al. titled "**Siamese Neural Networks for One-Shot Image Recognition**."

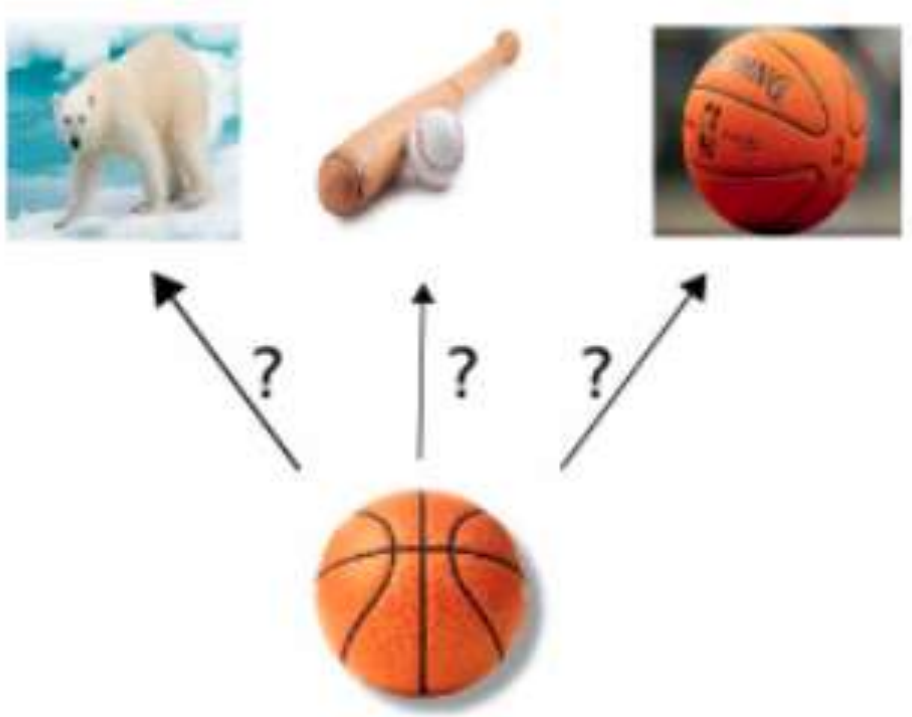
  https://www.cs.cmu.edu/~rsalakhu/papers/oneshot1.pdf

Siamese Neural Networks for One-Shot Image Recognition

same	"cow" (speaker #1)	"cow" (speaker #2)	same
different	"cow" (speaker #1)	"cat" (speaker #2)	different
same	"can" (speaker #1)	"can" (speaker #2)	same
different	"can" (speaker #1)	"cab" (speaker #2)	different

Verification tasks (training)

Siamese Neural Networks for One-Shot Image Recognition



Triplet loss

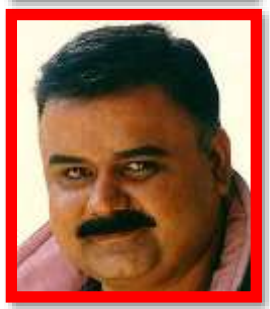
Triplet loss

Positive

minimize

Anchor

maximize



Negative

$$d(Anchor, Positive) + \alpha < d(Anchor, Negative)$$

[Schroff et al.,2015, FaceNet: A unified embedding for face recognition and clustering]

Triplet loss

Anchor
(A)

Positive
(p)

Anchor
(A)

Negative
(N)

[Schroff et al.,2015, FaceNet: A unified embedding for face recognition and clustering]

Triplet loss

Anchor
(A)

Positive
(p)

Anchor
(A)

Negative
(N)

$$\|f(A) - f(P)\|_2^2 \quad \leq \quad \|f(A) - f(N)\|_2^2$$

[Schroff et al.,2015, FaceNet: A unified embedding for face recognition and clustering]

Triplet loss

Anchor (A)

Positive (p)

Anchor (A)

Negative (N)

$$\underbrace{\|f(A) - f(P)\|_2^2}_{d(A,P)} \leq \underbrace{\|f(A) - f(N)\|_2^2}_{d(A,N)}$$

[Schroff et al.,2015, FaceNet: A unified embedding for face recognition and clustering]

Triplet loss

Anchor
(A)

Positive
(p)

Anchor
(A)

Negative
(N)

$$\|f(A) - f(P)\|_2^2 \leq \|f(A) - f(N)\|_2^2$$

$$\|f(A) - f(P)\|_2^2 - \|f(A) - f(N)\|_2^2 \leq 0$$

[Schroff et al.,2015, FaceNet: A unified embedding for face recognition and clustering]

Triplet loss

Anchor (A)

Positive (p)

Anchor (A)

Negative (N)

$$\|f(A) - f(P)\|_2^2 \leq \|f(A) - f(N)\|_2^2$$

$$\underbrace{\|f(A) - f(P)\|_2^2}_{0} - \underbrace{\|f(A) - f(N)\|_2^2}_{0} \leq 0$$

$$f(img) = \vec{0}$$

[Schroff et al.,2015, FaceNet: A unified embedding for face recognition and clustering]

Triplet loss

Anchor (A)

Positive (p)

Anchor (A)

Negative (N)

$$\|f(A) - f(P)\|_2^2 + \alpha \leq \|f(A) - f(N)\|_2^2$$

$$\underbrace{\|f(A) - f(P)\|_2^2}_{0} - \underbrace{\|f(A) - f(N)\|_2^2}_{0} + \alpha \leq 0$$

margin

$$f(img) = \vec{0}$$

[Schroff et al.,2015, FaceNet: A unified embedding for face recognition and clustering]

Triplet loss

Anchor (A)

Positive (p)

Anchor (A)

Negative (N)

$$\underbrace{\|f(A)-f(P)\|_2^2}_{d(A,P)} + \alpha \leq \underbrace{\|f(A)-f(N)\|_2^2}_{d(A,N)}$$

$d(A,P) = 0.6$ $\quad$ 0.2

$$\underbrace{\|f(A)-f(P)\|_2^2}_{0} - \underbrace{\|f(A)-f(N)\|_2^2}_{0} + \alpha \leq 0$$

margin

$$f(img) = \vec{0}$$

[Schroff et al.,2015, FaceNet: A unified embedding for face recognition and clustering]

Triplet loss

Anchor (A)

Positive (p)

Anchor (A)

Negative (N)

$$\underbrace{\|f(A) - f(P)\|_2^2}_{d(A,P)} + \alpha \leq \underbrace{\|f(A) - f(N)\|_2^2}_{d(A,N)}$$

$d(A,P) = 0.6$ $\quad 0.2$ $\quad d(A,N) = 0.63$

$$\underbrace{\|f(A) - f(P)\|_2^2}_{0} - \underbrace{\|f(A) - f(N)\|_2^2}_{0} + \alpha \leq 0$$

margin

$$f(img) = \vec{0}$$

[Schroff et al.,2015, FaceNet: A unified embedding for face recognition and clustering]

Triplet loss

Anchor (A)

Positive (p)

Anchor (A)

Negative (N)

$$\underbrace{\|f(A) - f(P)\|_2^2}_{d(A,P)} + \alpha \leq \underbrace{\|f(A) - f(N)\|_2^2}_{d(A,N)}$$

$d(A,P) = 0.6$ 0.2 $d(A,N) = 0.63$ 0.8

$$\underbrace{\|f(A) - f(P)\|_2^2}_{0} - \underbrace{\|f(A) - f(N)\|_2^2}_{0} + \alpha \leq 0$$

margin

$$f(img) = \vec{0}$$

[Schroff et al.,2015, FaceNet: A unified embedding for face recognition and clustering]

تابع loss

- با فرض داشتن A, P و N

$$\mathcal{L}(A, P, N) = \|f(A) - f(P)\|^2 - \|f(A) - f(N)\|^2 + \alpha$$

[Schroff et al.,2015, FaceNet: A unified embedding for face recognition and clustering]

تابع loss

- با فرض داشتن A, P و N

$$\mathcal{L}(A, P, N) = max(\|f(A) - f(P)\|^2 - \|f(A) - f(N)\|^2 + \alpha, 0)$$

[Schroff et al.,2015, FaceNet: A unified embedding for face recognition and clustering]

تابع loss

- با فرض داشتن A, P و N

$$\mathcal{L}(A, P, N) = max(\|f(A) - f(P)\|^2 - \|f(A) - f(N)\|^2 + \alpha, 0)$$

$$J = \sum_{i=1}^{m} \mathcal{L}\left(A^{(i)}, P^{(i)}, N^{(i)}\right)$$

[Schroff et al.,2015, FaceNet: A unified embedding for face recognition and clustering]

تابع loss

- با فرض داشتن A, P و N

$$\mathcal{L}(A, P, N) = max(\|f(A) - f(P)\|^2 - \|f(A) - f(N)\|^2 + \alpha, 0)$$

$$J = \sum_{i=1}^{m} \mathcal{L}\left(A^{(i)}, P^{(i)}, N^{(i)}\right)$$

- اگر دیتاستی با ۱۰.۰۰۰ تصویر از ۱۰۰۰ فرد داشته باشیم باید از ۱۰۰۰۰ تصویر برای ساخت سه تایی ها استفاده کنیم.

[Schroff et al.,2015, FaceNet: A unified embedding for face recognition and clustering]

انتخاب سه‌تایی‌ها(Tripletها) A و P و N

- اگر A و P و N به صورت تصادفی انتخاب شوند شرط

$$d(A, \mathrm{P}) + \alpha \leq d(A, N)$$

به راحتی برآورده می‌شود.

- انتخاب Tripletهایی که برای آموزش «سخت» است.

$$d(A, \mathrm{P}) + \alpha \leq d(A, N)$$

$$d(A, \mathrm{P}) \approx d(A, N)$$

[Schroff et al.,2015, FaceNet: A unified embedding for face recognition and clustering]

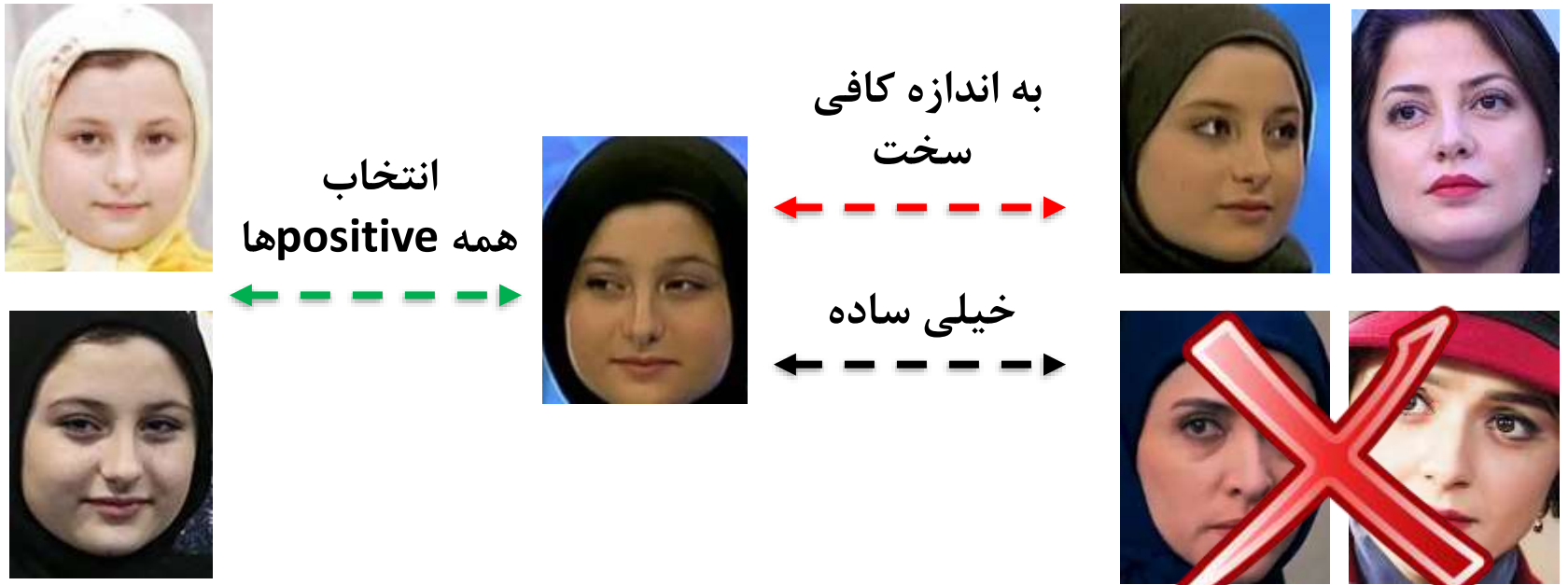
مجموعه داده آموزش(Training set) برای Triplet Loss

Anchor	Positive	Negative
		
		
⋮	⋮	⋮
		

انتخاب Tripletها

- مشکل اصلی:
  - چگونه سه‌تایی‌ها را انتخاب کنیم؟
- راه حل؟
  - انتخاب Tripleهای سخت در mini-batch بزرگ (>۱۰۰۰)

Useful triplets = hardest errors



Choosing triplets: trap

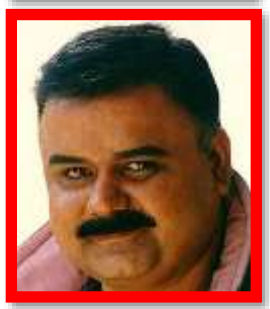
انتخاب Tripletها: تله!

Positive

minimize

Anchor

maximize



Negative

$$Positive \approx Negative$$

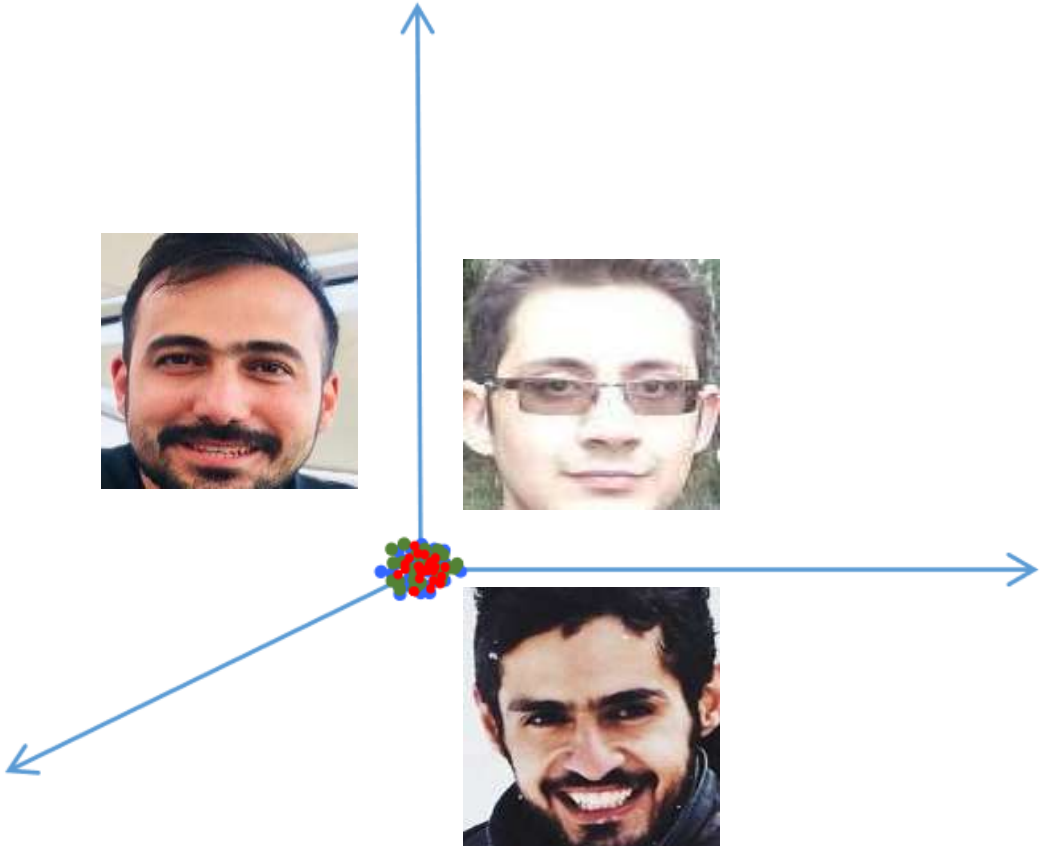
[Schroff et al.,2015, FaceNet: A unified embedding for face recognition and clustering]

انتخاب Tripletها: تله!

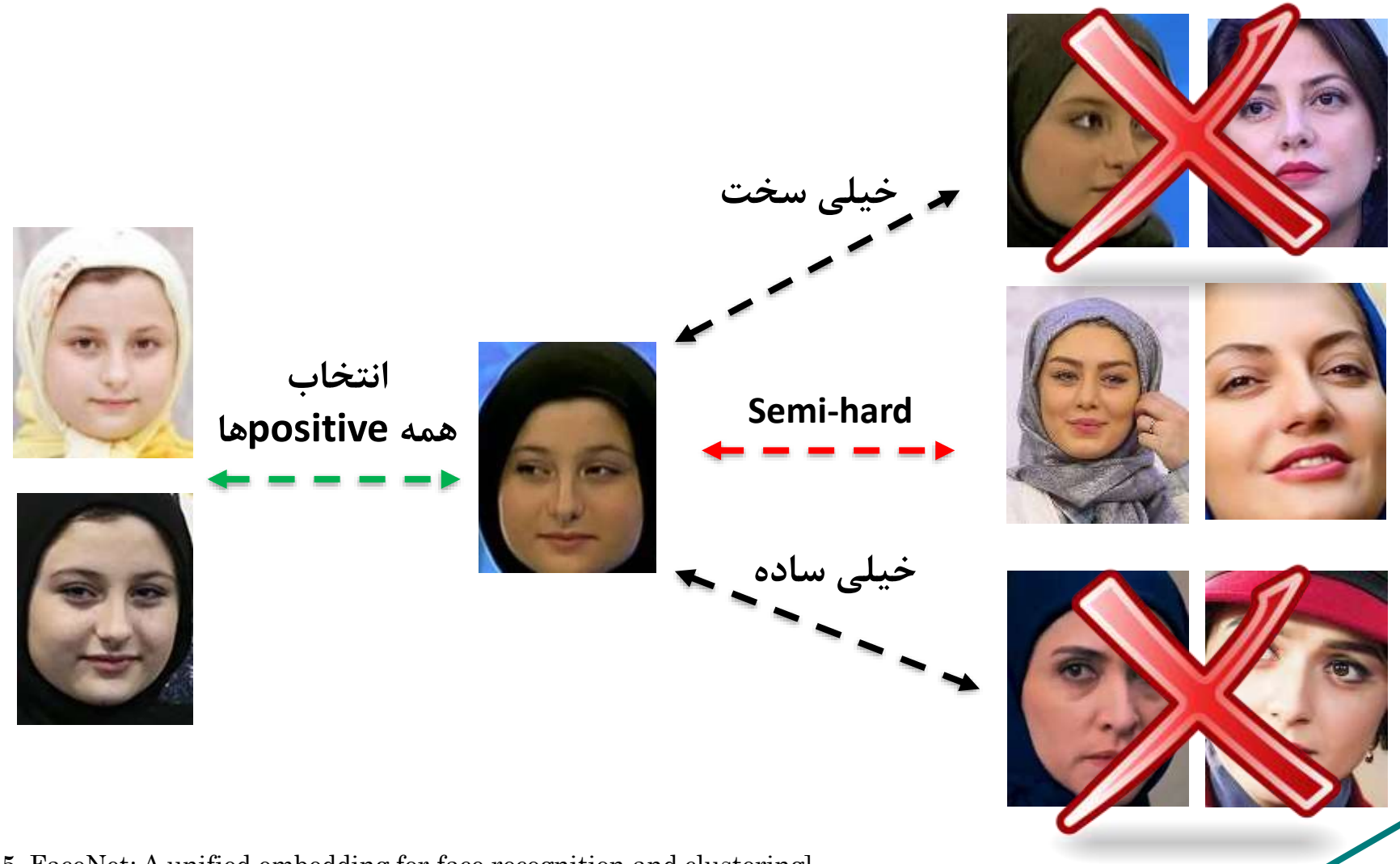
به جای ...

انتخاب Tripletها: تله!

- Selecting hardest negative may lead to the collapse early in training

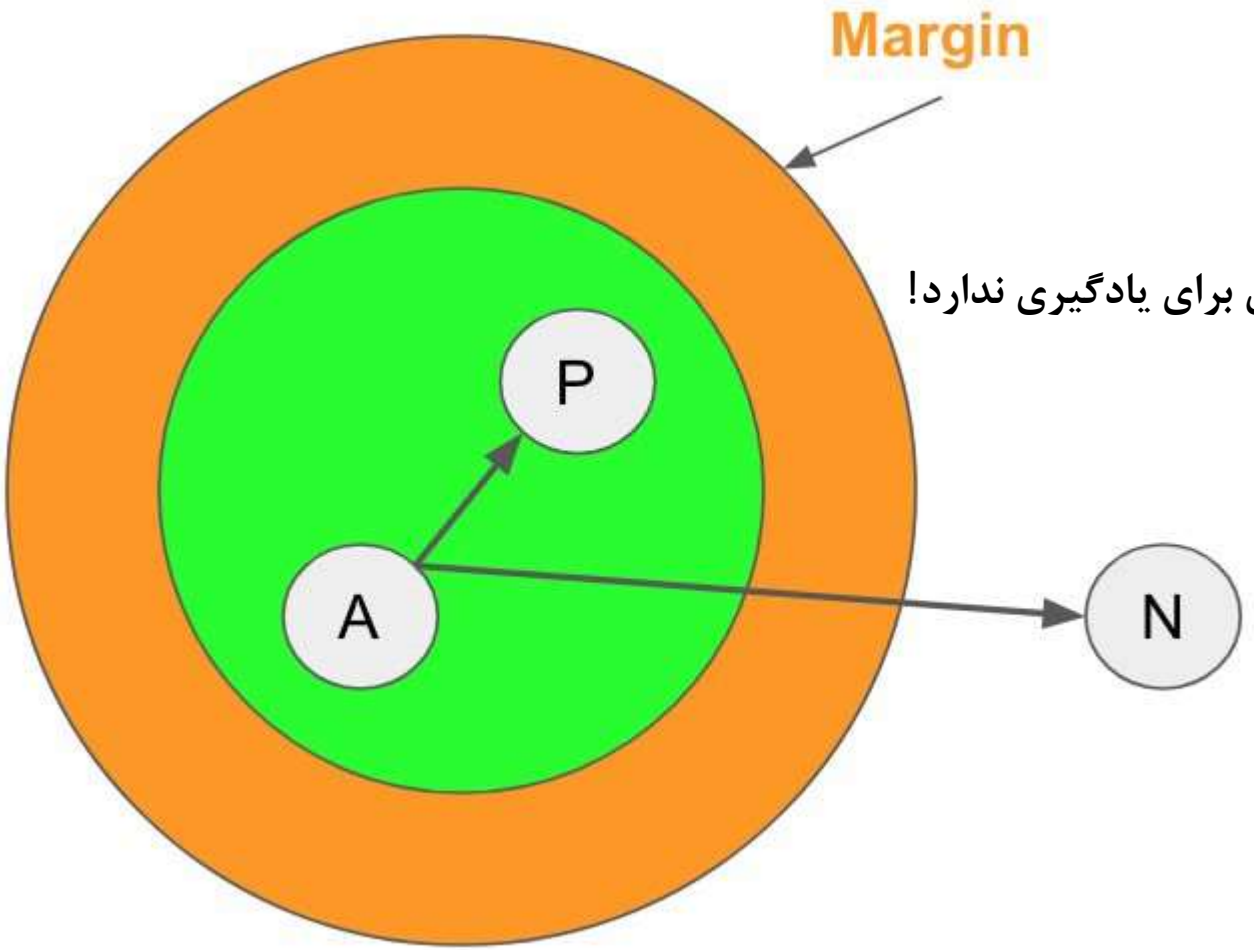


انتخاب Tripletها



[Schroff et al.,2015, FaceNet: A unified embedding for face recognition and clustering]

انتخاب Tripletها

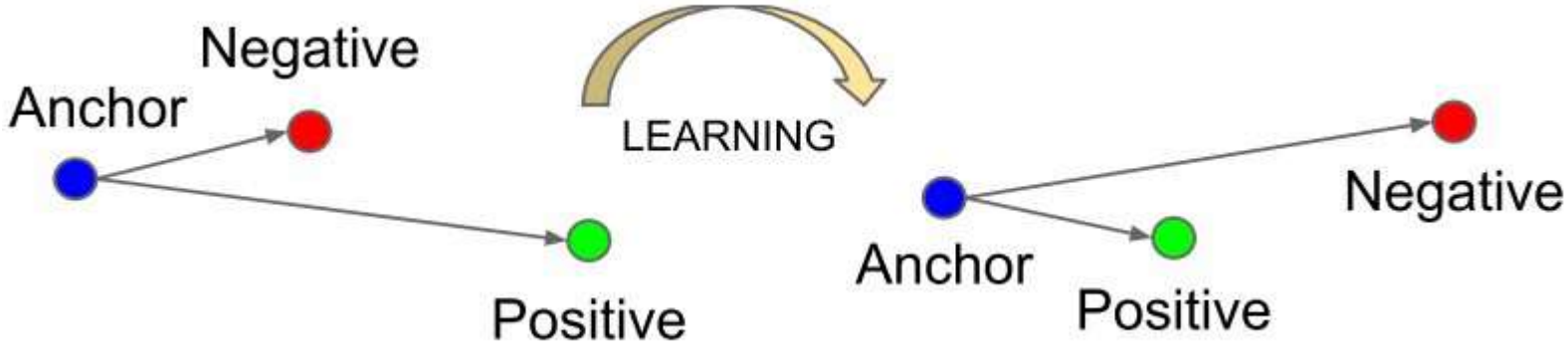


- ✓ خیلی ساده! شرایط از قبل برقرار است و چیزی برای یادگیری ندارد!

$$d(Anchor, Positive) + \alpha < d(Anchor, Negative)$$

[Schroff et al.,2015, FaceNet: A unified embedding for face recognition and clustering]

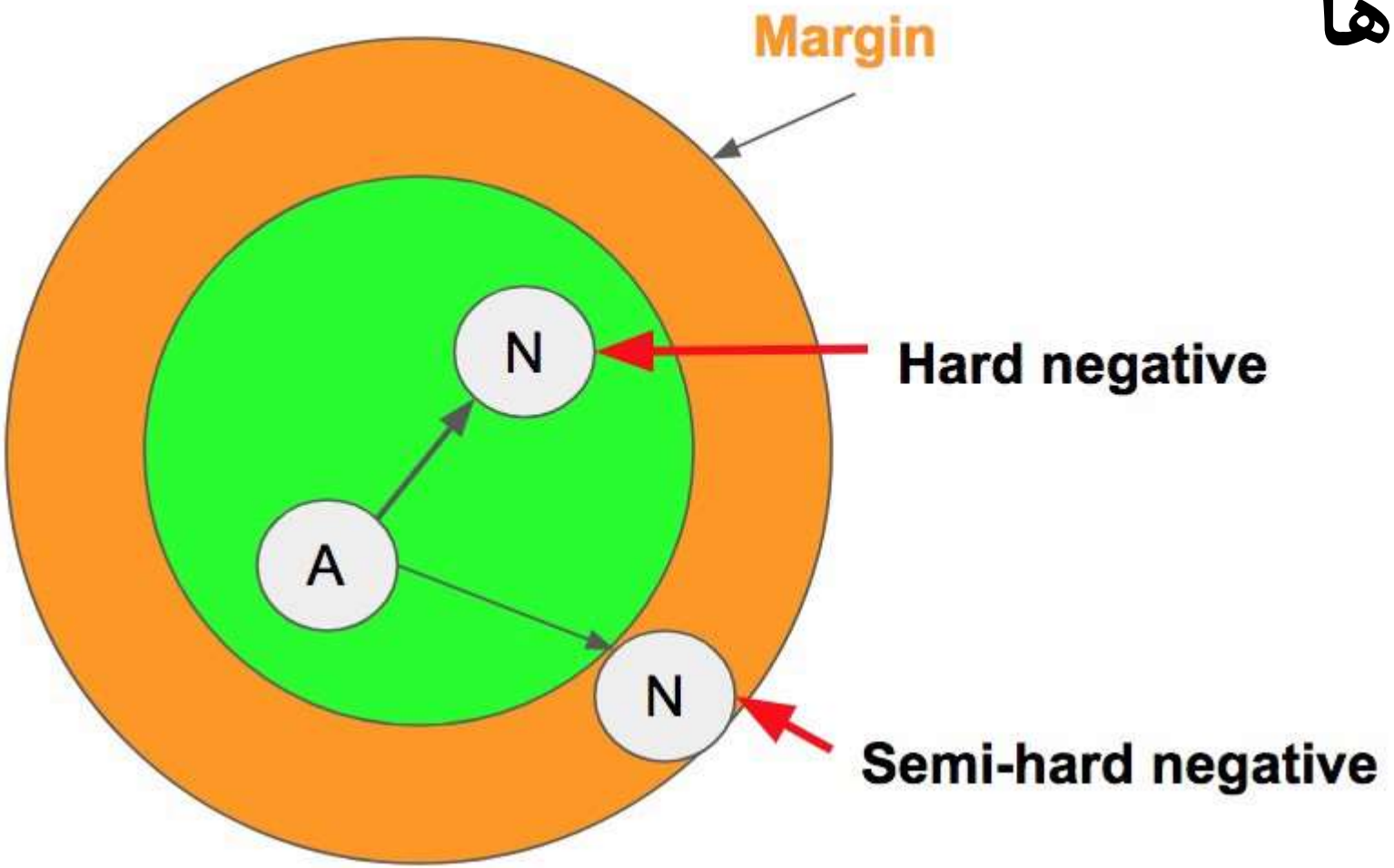
انتخاب Tripletها



$$d(Anchor, Positive) + \alpha < d(Anchor, Negative)$$

[Schroff et al.,2015, FaceNet: A unified embedding for face recognition and clustering]

انتخاب Tripletها



$$d(Anchor, Positive) + \alpha < d(Anchor, Negative)$$

[Schroff et al.,2015, FaceNet: A unified embedding for face recognition and clustering]

Let’s code…

- 2-losses_triplet.ipynb

  http://colab.research.google.com/github/Alireza-Akhavan/deep-face-recognition/blob/master/aaiss2020/2-losses_triplet.ipynb

More…

- … for models trained from scratch as well as pretrained ones, using a variant of the triplet loss to perform end-to-end deep metric learning outperforms most other published methods by a large margin.

— In Defense of the Triplet Loss for Person Re-Identification, 2017.

https://arxiv.org/abs/1703.07737

خلاصه Triplet loss

- خلاصه:
  - نیاز به سایز mini-batch بزرگ و پیدا کردن margin (α)
  - همگرایی کند و آهسته
- پیاده سازی‌ها(غیر رسمی):
  - OpenFace(Torch)

https://github.com/cmusatyalab/openface

  - davidsandberg/facenet(Tensorflow)

https://github.com/davidsandberg/facenet

	LFW	Megaface
Google’s Facenet	99.63	70.5

...

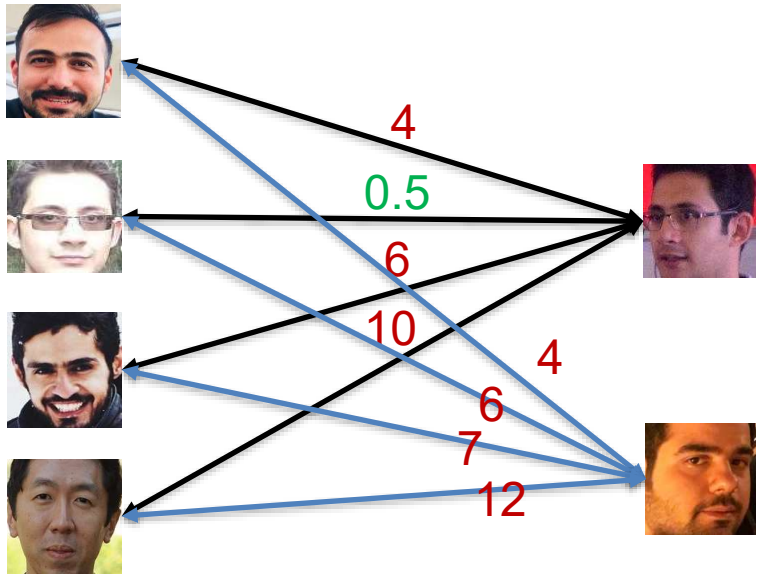
یادگیری تابع «شباهت» (“similarity” function)

- d(img1,img2) = میزان تفاوت بین دو تصویر

If $d(img1,img2) \leq \tau$: یکسان

$> \tau$: متفاوت

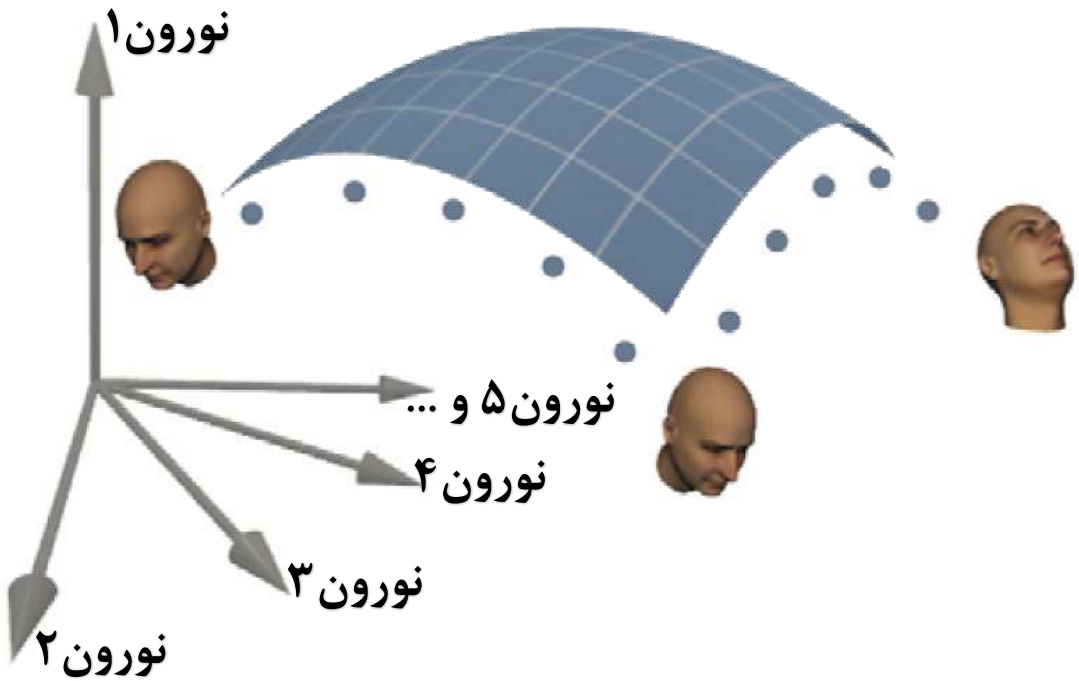
} Face Verification



«علیرضا اخوان پور»

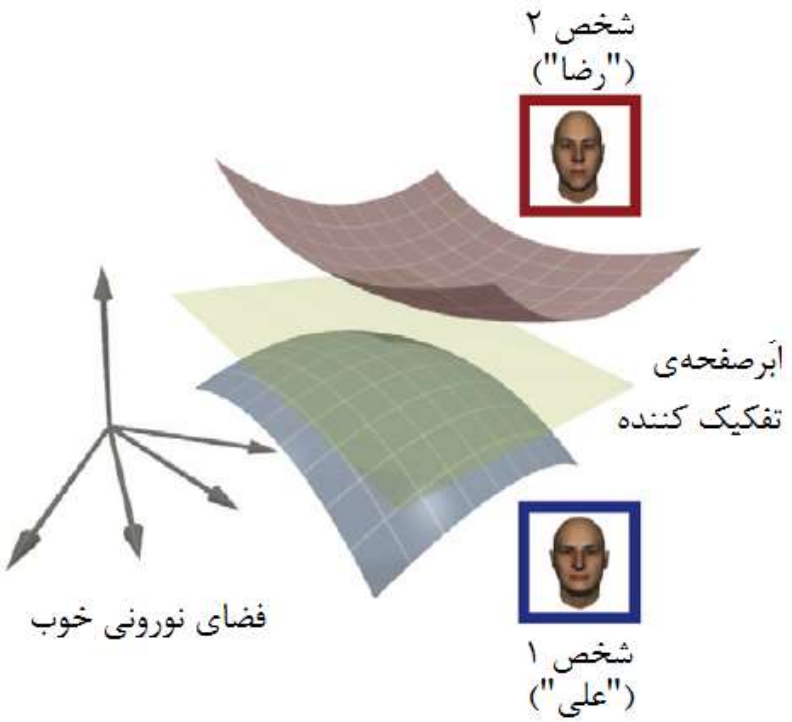
تصویر فرد ناشناس

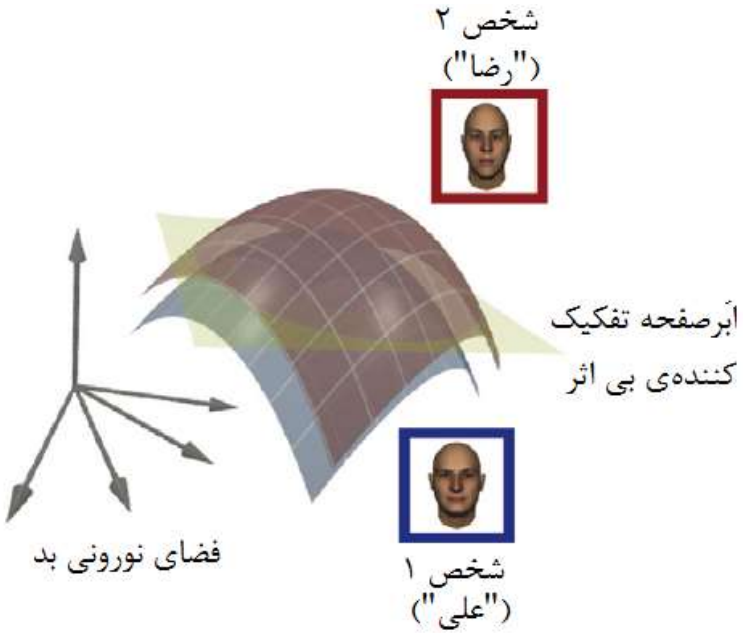
فضای نرونی و خمینه (manifold)



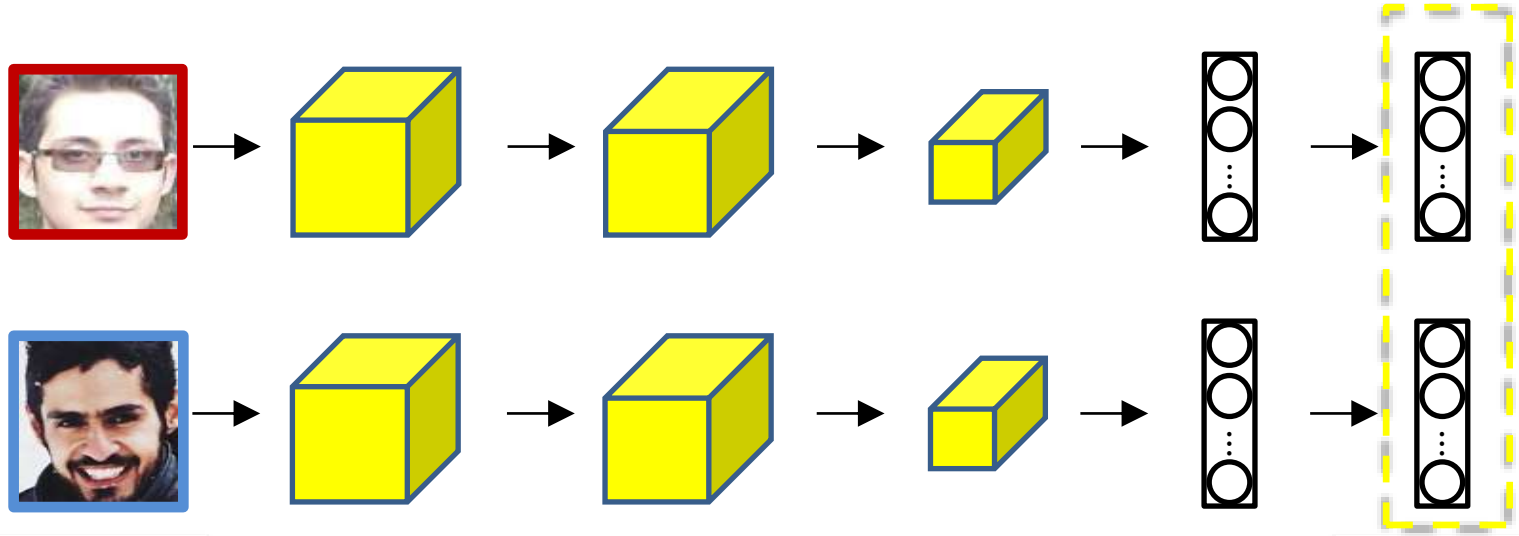
*DiCarlo and Cox , **TICS** (2007)*

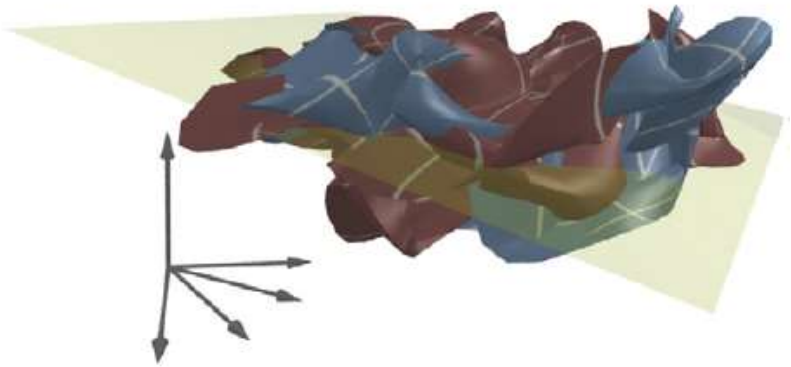
فضای نرونی خوب و بد





*DiCarlo and Cox , **TICS** (2007)*

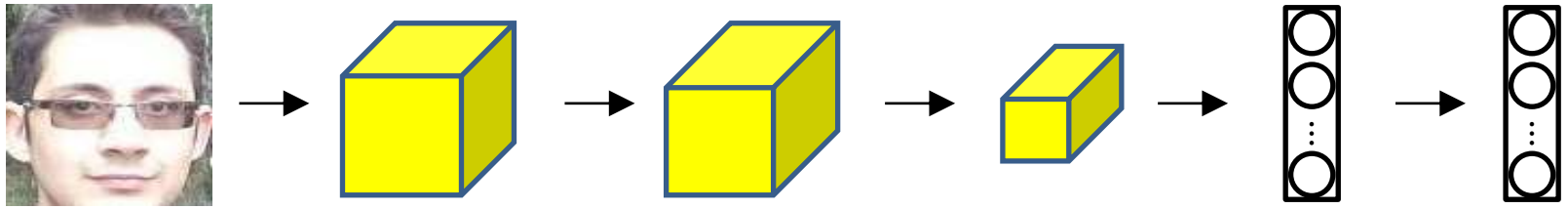




فضای واقعی پیکسل

فضای 128تایی encoding تصاویر

هدف یادگیری



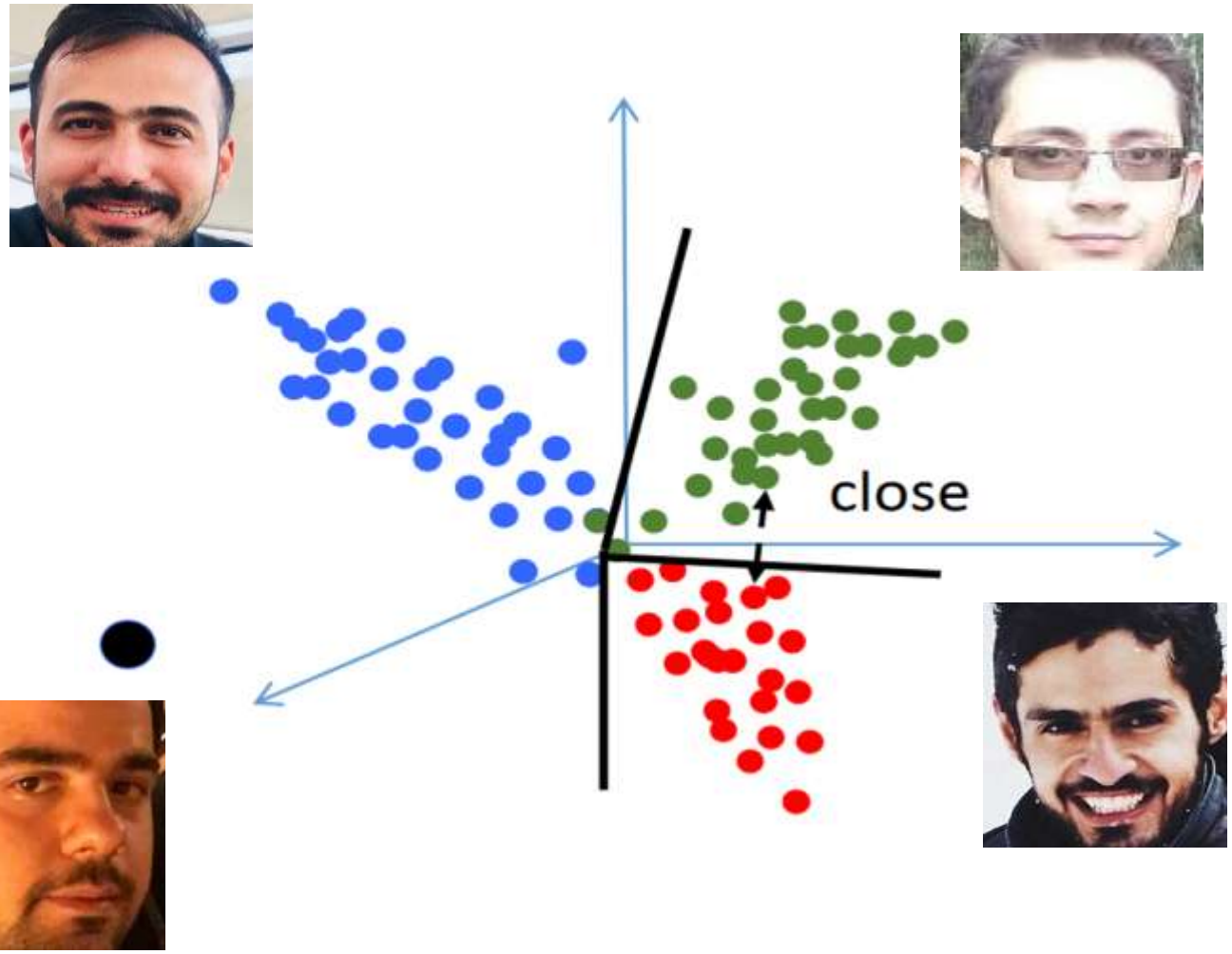
- پارامترهای شبکه عصبی encoding خروجی یا $f(x^{(i)})$ را تشکیل می‌دهند
- یادگیری پارامترها به نحوی که:
  - اگر $x^{(i)}, x^{(j)}$ دو فرد یکسان بودند، $\|f(x^{(i)}) - f(x^{(j)})\|^2$ مقدار کوچکی شود.
  - اگر $x^{(i)}, x^{(j)}$ دو فرد متفاوت بودند، $\|f(x^{(i)}) - f(x^{(j)})\|^2$ مقدار بزرگی شود.

بخش ۲:

یادگیری متریک

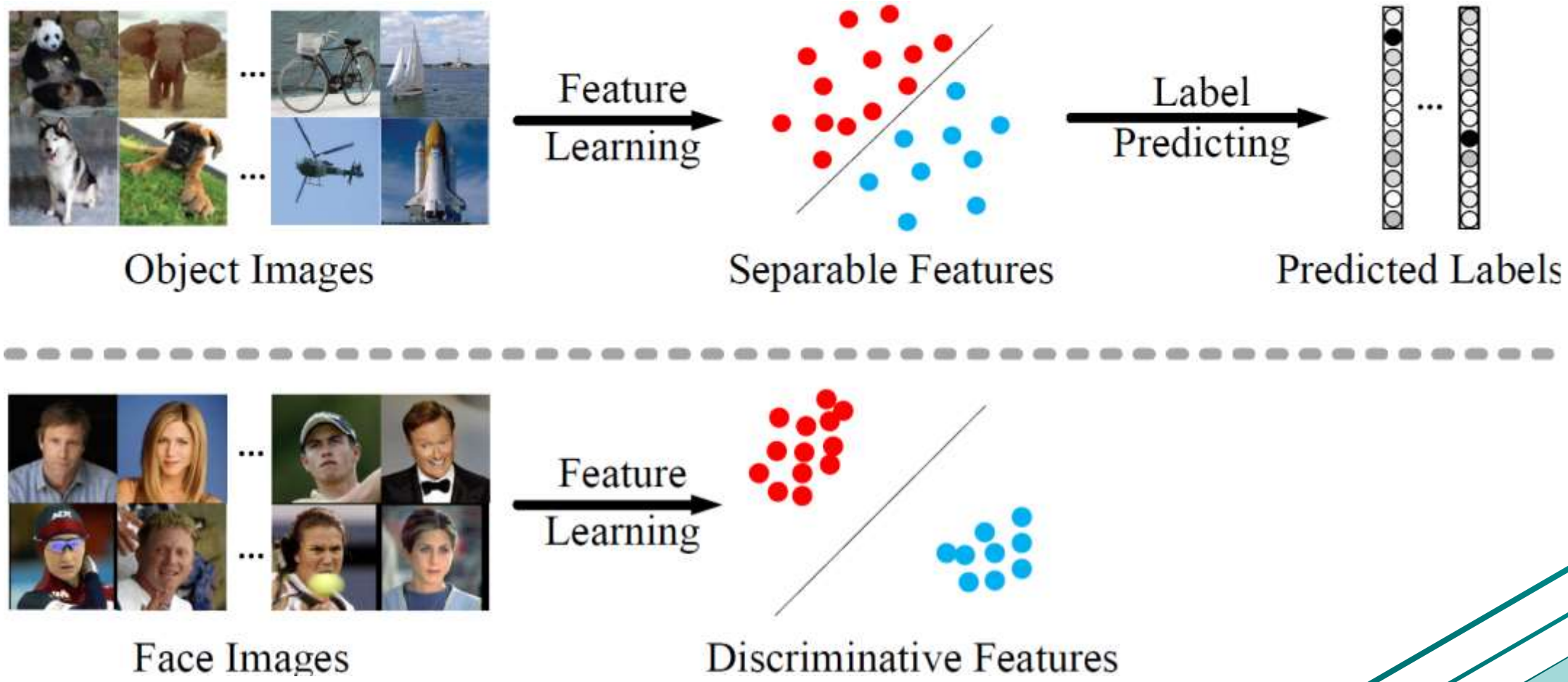
Metric learning

تابع Softmax



- ویژگی‌هایی که آموزش می‌بیند فقط separable هستند، اما discriminative نیستند!
- این ویژگی‌ها به اندازه کافی موثر نیستند.

نیاز به metric learning داریم!

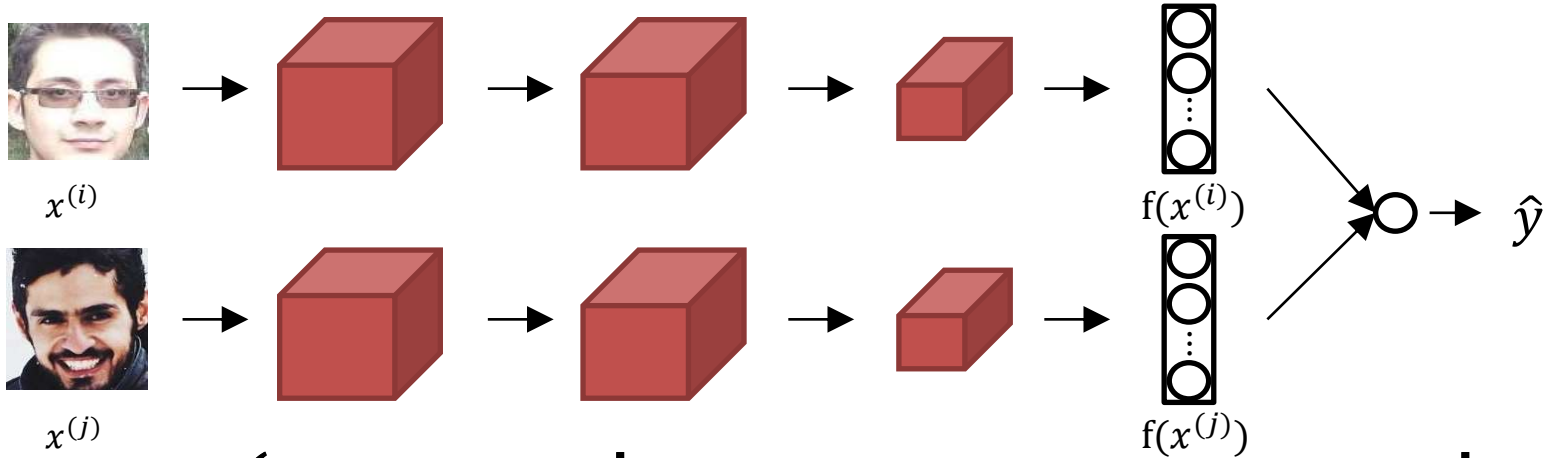


[Yandong Wen et. al., 2016. A Discriminative Deep Feature Learning Approach for Face Recognition]

نیاز به metric learning داریم!

- Tightness of the cluster❑
- Discriminative features❑

Learning the similarity function



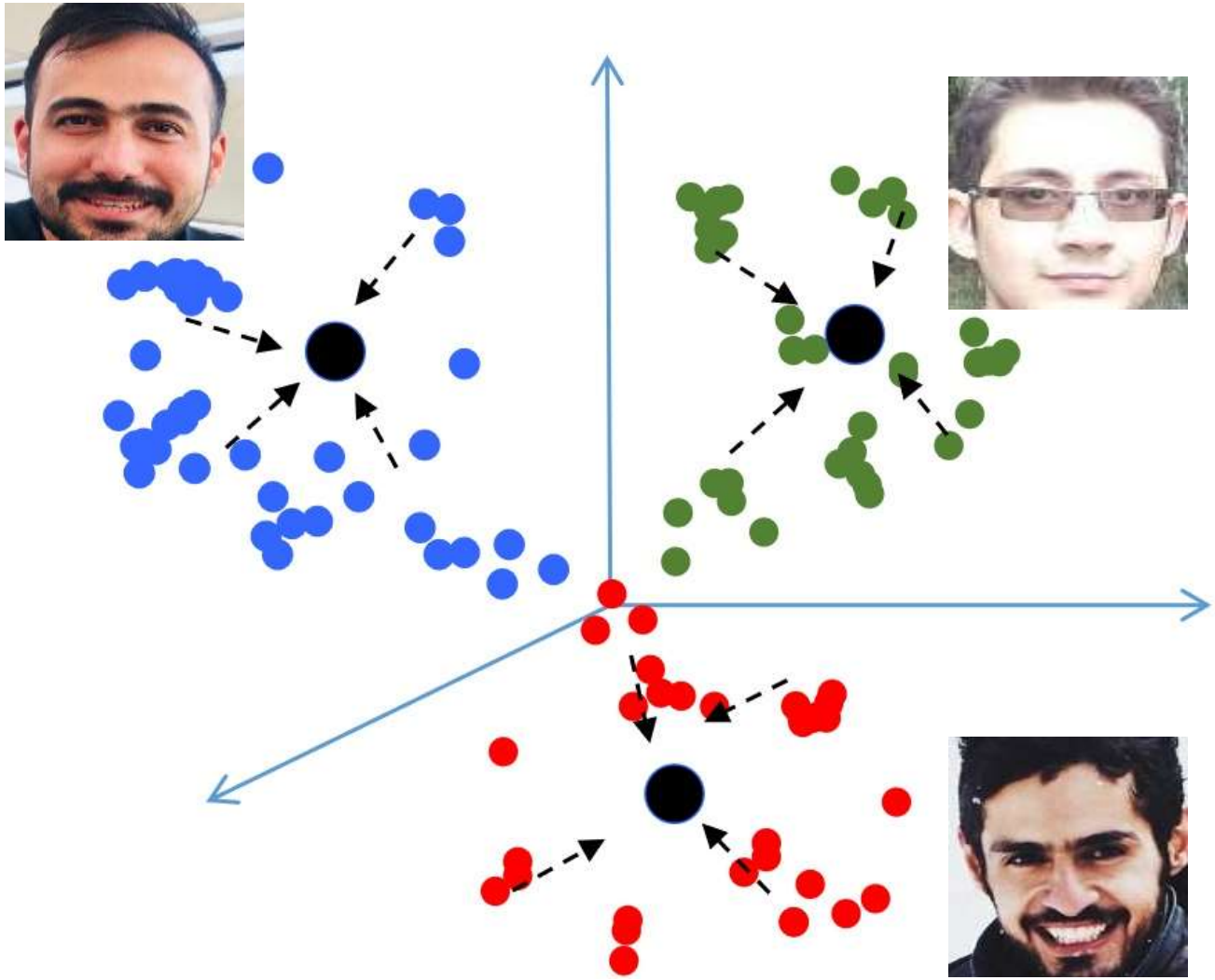
$$\hat{y} = \sigma\left(\sum_{k=1}^{128} w_i \left| f\left(x^{(i)}\right)_k - f\left(x^{(j)}\right)_k \right| + b\right)$$

$$\frac{\left(f\left(x^{(i)}\right)_k - f\left(x^{(j)}\right)_k\right)^2}{f(x^{(i)})_k + f(x^{(j)})_k} \quad \chi^2$$

[Taigman et. al., 2014. DeepFace closing the gap to human level performance]

Center loss

Center Loss



- **Idea:** pull the points to class centroids

[Wen et al.,2016, A discriminative feature learning approach for deep face recognition]

Separable vs Discriminative

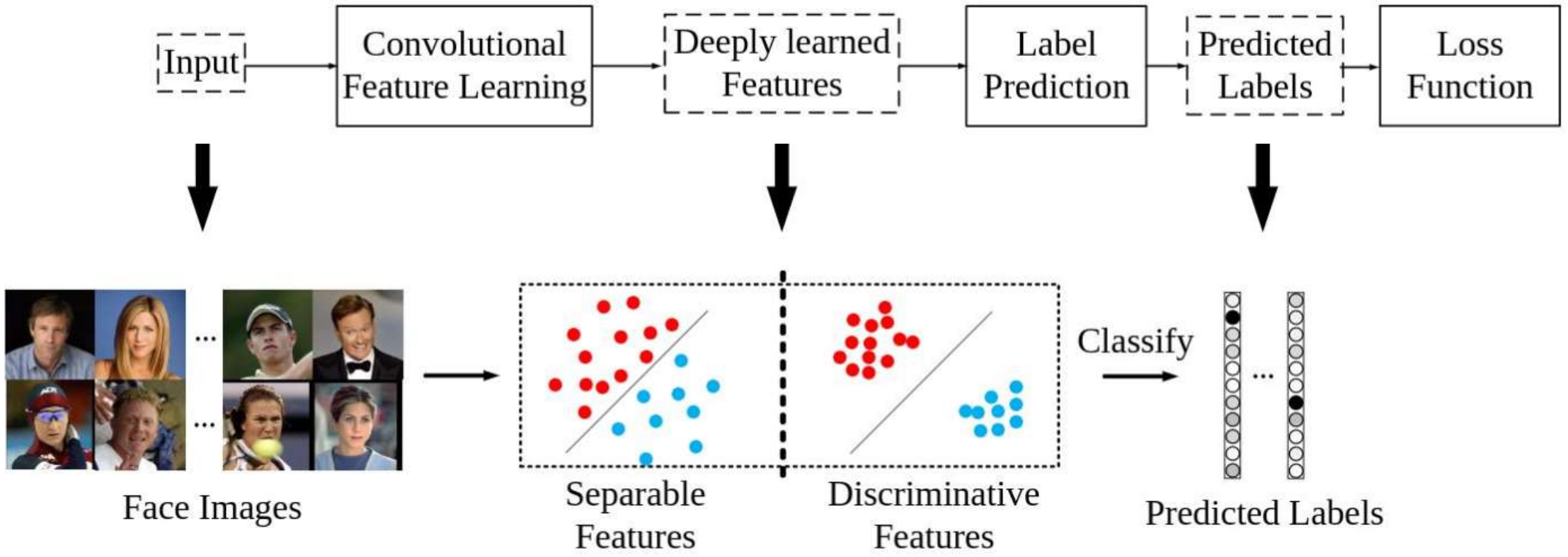


Fig. 1. The typical framework of convolutional neural networks.

[Wen et al.,2016, A discriminative feature learning approach for deep face recognition]

A toy example : What's wrong with Softmax

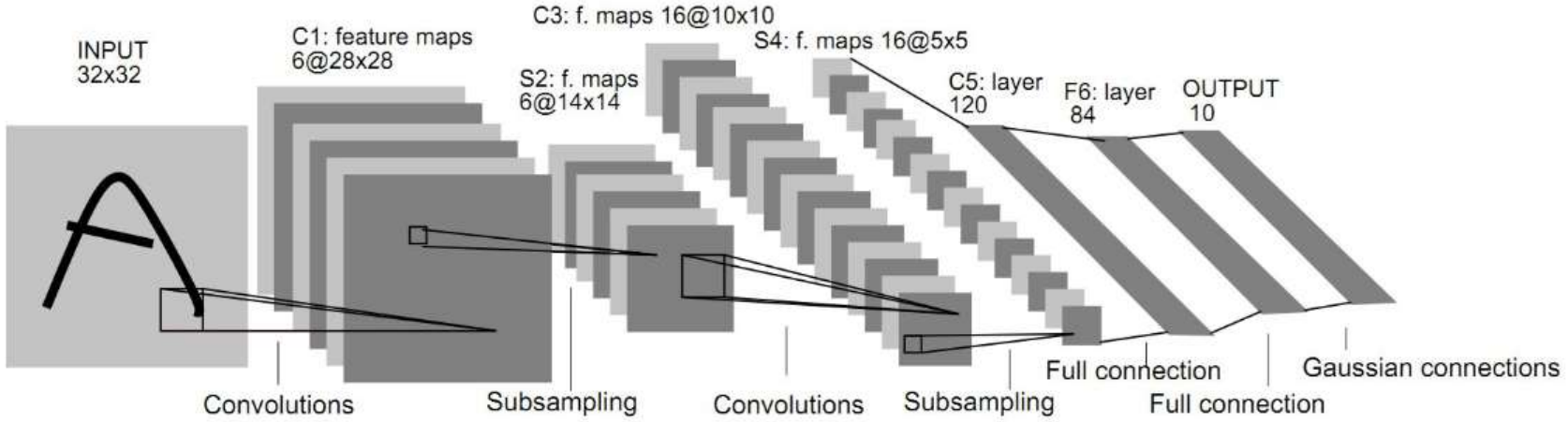


Fig. 2. Architecture of LeNet-5, a Convolutional Neural Network, here for digits recognition. Each plane is a feature map, i.e. a set of units whose weights are constrained to be identical.

	stage 1		stage 2		stage 3		stage 4
Layer	conv	pool	conv	pool	conv	pool	FC
LeNets	$(5,20)_{/1,0}$	$2_{/2,0}$	$(5,50)_{/1,0}$	$2_{/2,0}$			500
LeNets++	$(5,32)_{/1,2} \times 2$	$2_{/2,0}$	$(5,64)_{/1,2} \times 2$	$2_{/2,0}$	$(5,128)_{/1,2} \times 2$	$2_{/2,0}$	**2**

[Wen et al.,2016, A discriminative feature learning approach for deep face recognition]

A toy example : What's wrong with Softmax

$\boldsymbol{m}$: mini-batch size

$\boldsymbol{n}$: the number of class

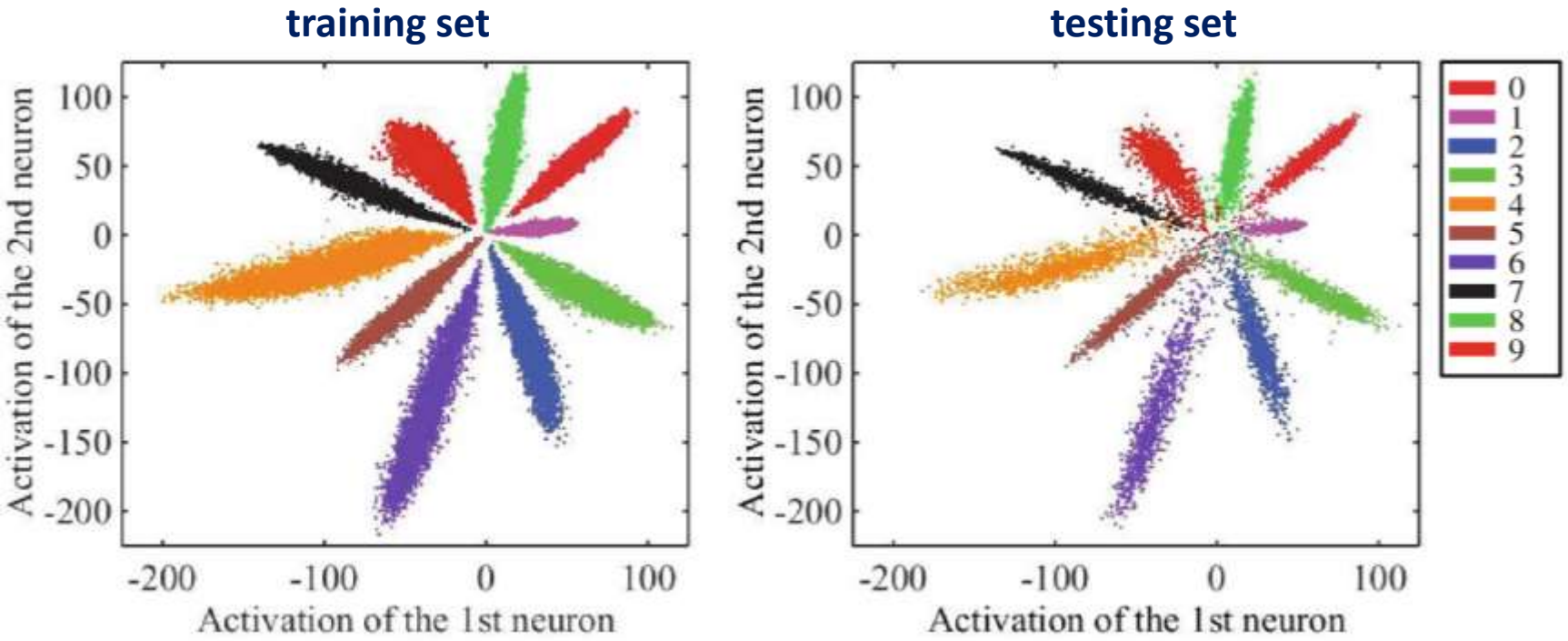
$\boldsymbol{x_i}$: feature vector in R^d (d is the feature dimension)

$\boldsymbol{w}$: $R^{(d\times n)}$, $\boldsymbol{b}$: R^n(bias)

$$\mathcal{L}_s = -\sum_{i=1}^{m} \log \frac{e^{w_{y_i}^T x_i + b_{y_i}}}{\sum_{j=1}^{n} e^{w_j^T x_i + b_j}}$$

[Wen et al.,2016, A discriminative feature learning approach for deep face recognition]

آیا ویژگی‌ها برای خوشه بندی مناسب است؟



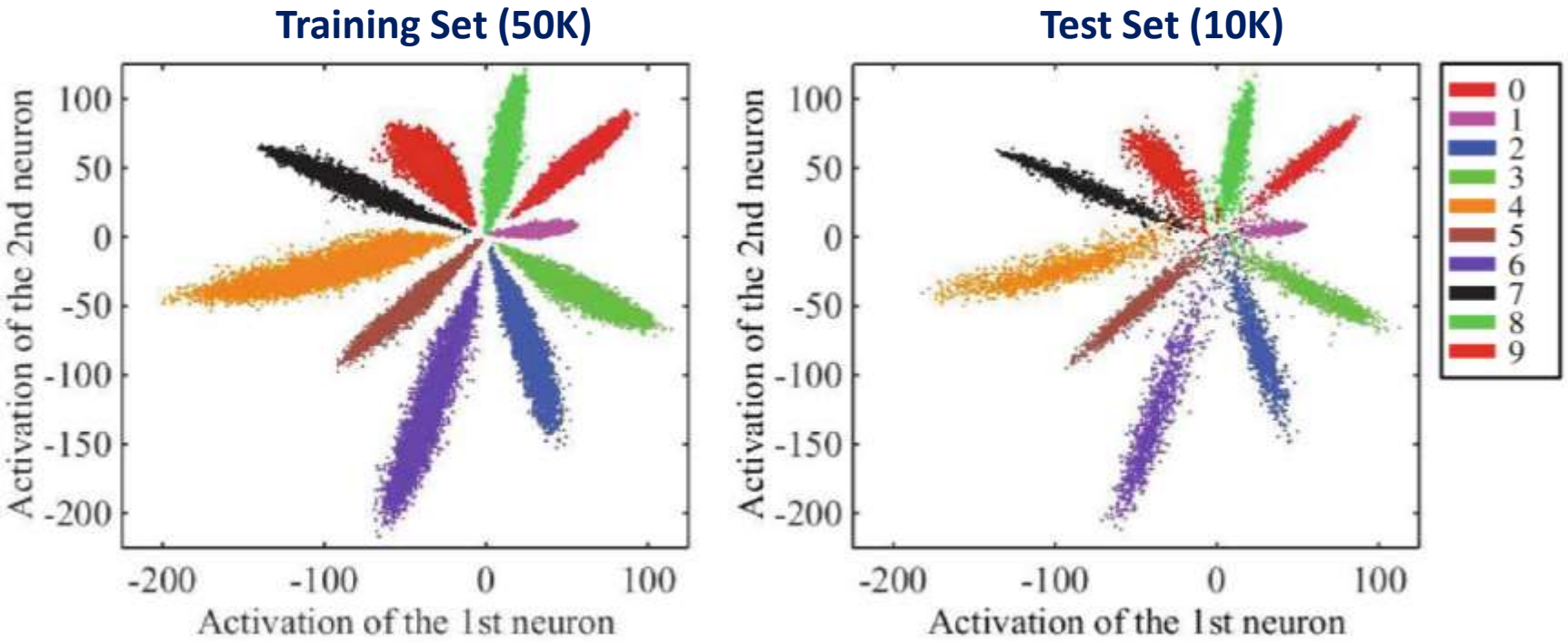
Separable,

the deep features are not discriminative enough.

by intra-class variation

[Wen et al.,2016, A discriminative feature learning approach for deep face recognition]

آیا ویژگی‌ها برای خوشه بندی مناسب است؟



Separable,

the deep features are not discriminative enough.

by intra-class variation

[Wen et al.,2016, A discriminative feature learning approach for deep face recognition]

افزودن center loss برای discriminative شدن ویژگی‌ها!

$\boldsymbol{m}$: mini-batch size

$\boldsymbol{C_{y_i}}$: y_{th} class center in d dimension

$\boldsymbol{x_i}$: feature vector in R^d (d is the feature dimension)

- اما! $\boldsymbol{C_{y_i}}$ با آموزش شبکه و تغییرات ویژگی‌ها باید به روز شود.
- میانگین ویژگی‌های هر کلاس در هر iteration

$$\mathcal{L}_C = \frac{1}{2}\sum_{i=1}^{m}\|x_i - C_{y_i}\|_2^2$$

[Wen et al.,2016, A discriminative feature learning approach for deep face recognition]

تابع نهایی $\mathcal{L} = \mathcal{L}_s + \lambda \mathcal{L}_C$

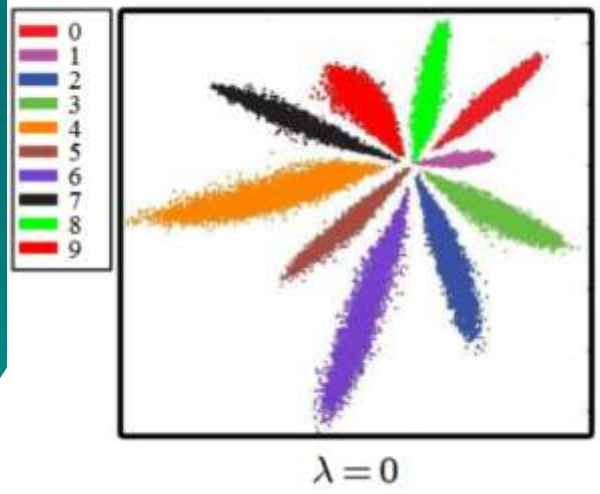
- **SOFTMAX LOSS: encouraging the separability of features.**
- **CENTER LOSS: simultaneously learning a center for deep features of each class and penalizing the distances between the deep features and their corresponding class centers.**
- **JOINT SUPERVISION: minimizing the intra-class variations while keeping the features of different classes separable.**

$$\mathcal{L} = \mathcal{L}_S + \lambda \mathcal{L}_C$$

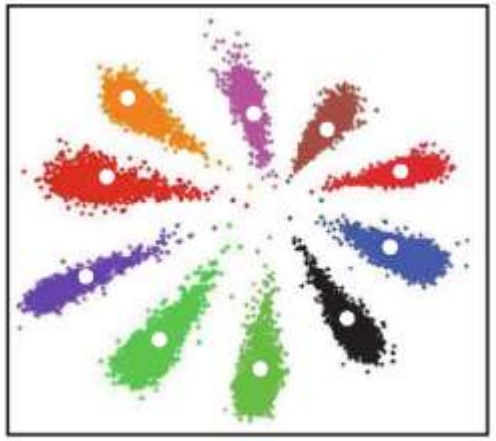
$$= -\sum_{i=1}^{m} \log \frac{e^{W_{y_i}^T \boldsymbol{x}_i + b_{y_i}}}{\sum_{j=1}^{n} e^{W_j^T \boldsymbol{x}_i + b_j}} + \frac{\lambda}{2} \sum_{i=1}^{m} \| \boldsymbol{x}_i - \boldsymbol{c}_{y_i} \|_2^2$$

Inter-class Separability **Intra-class Compactness**

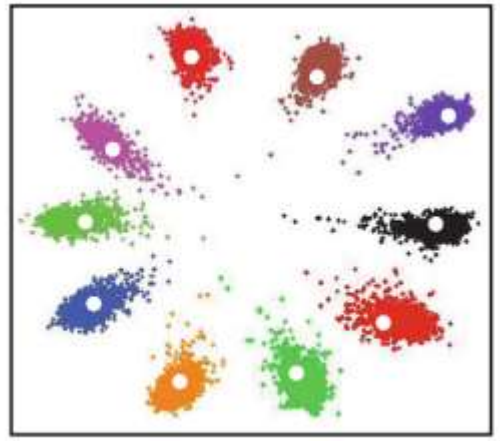
تابع نهایی $\mathcal{L} = \mathcal{L}_s + \lambda \mathcal{L}_C$ و تاثیر λ



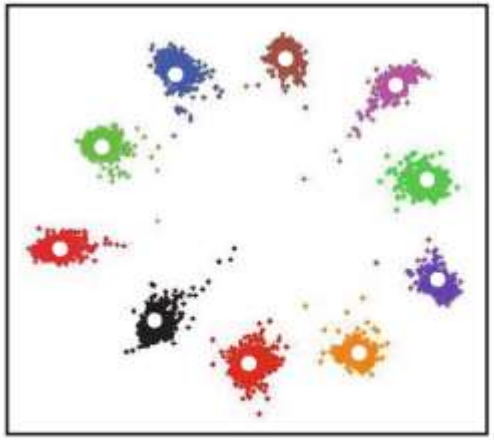
$\lambda = 0$



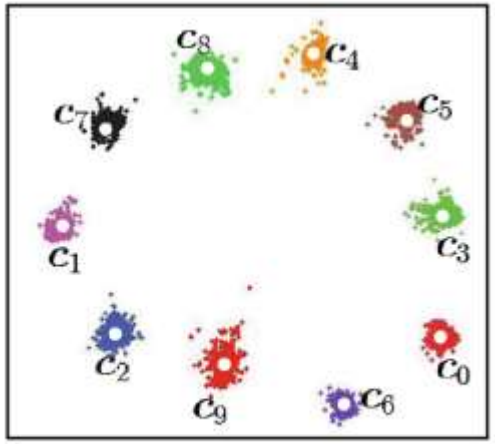
(a) $\lambda = 0.001$



(b) $\lambda = 0.01$



(c) $\lambda = 0.1$



(d) $\lambda = 1$

[Wen et al.,2016, A discriminative feature learning approach for deep face recognition]

Let’s code…

- 3_keras_center_loss_with_embedding.ipynb
  - http://colab.research.google.com/github/Alireza-Akhavan/deep-face-recognition/blob/master/aaiss2020/3_keras_center_loss_with_embedding.ipynb

خلاصه center loss

- خلاصه:
  - ایجاد توده‌ی درون کلاسی و جدایی بین کلاس‌ها
  - کارایی خوب، همگرایی ساده
- پیاده سازی‌ها:
  - (caffe)ydwen/caffe-face – کد رسمی منتشر شده توسط نویسنده مقاله
    https://github.com/ydwen/caffe-face
  - (Tensorflow)davidsandberg/facenet
    https://github.com/davidsandberg/facenet

	LFW	Megaface
Center Loss	99.28%	65.234%

مقایسه نتایج

	LFW	Megaface
Google’s Facenet	99.63	70.5
Center Loss	99.28%	65.234%

- بین 100M-200M تصویر چهره از ۸ میلیون شخص متفاوت
- ۰.۷ میلیون تصویر چهره از 17,189 شخص مختلف
- برای چالش مگافیس تنها از 490 هزار تصویر برای اموزش استفاده شده است.

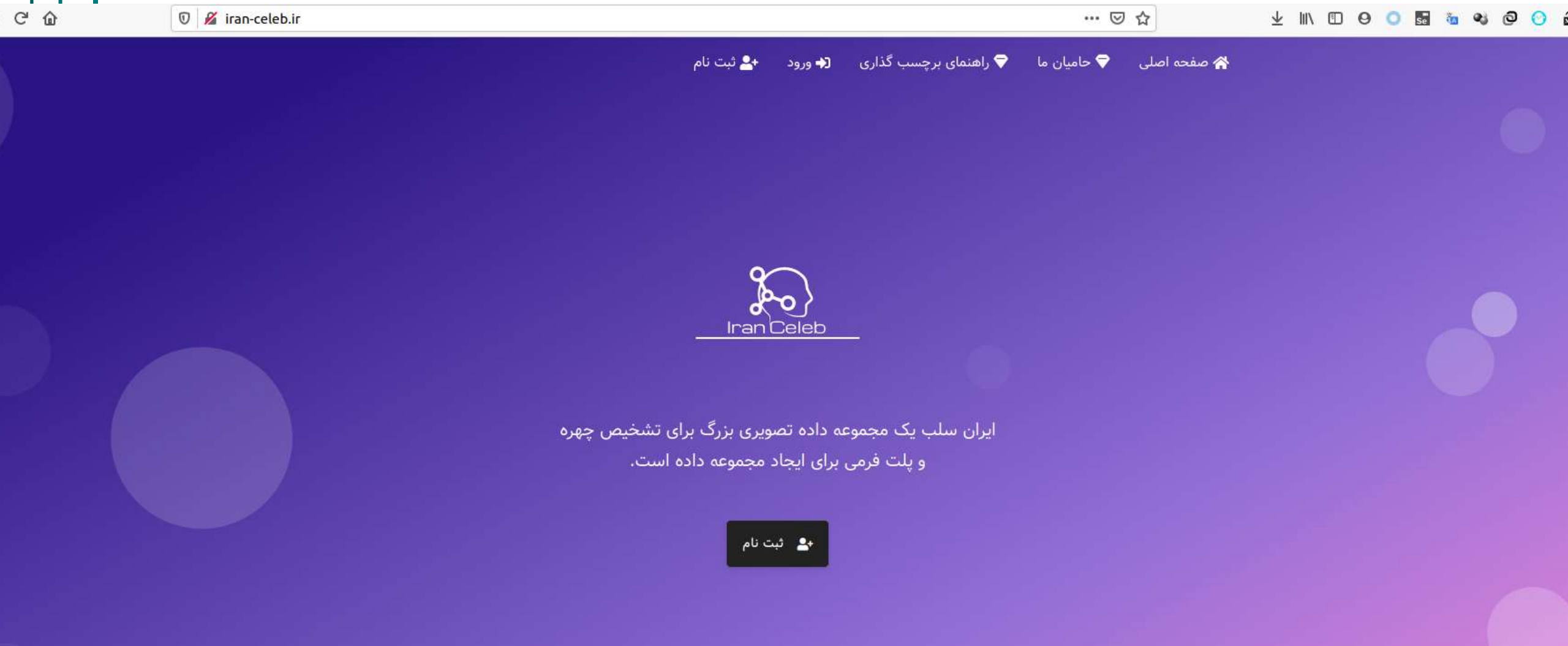
- **MegaFace**
  - **— Our model is trained on 490K face images, termed as model C-.**

Method	Protocol	Identification Acc. (Set 1)	Verification Acc. (Set 1)
NTechLAB - facenx_large	large	73.300%	85.081%
Google - FaceNet v8		70.496%	86.473%
Beijing Faceall Co. - FaceAll_Norm_1600		64.803%	67.118%
Beijing Faceall Co. - FaceAll_1600		63.977%	63.960%
Barebones_FR - cnn	small	59.363%	59.036%
NTechLAB - facenx_small		58.218%	66.366%
3DiVi Company – tdvm6		33.705%	36.927%
model A-	small	41.863%	41.297%
Model B-		57.175%	69.897%
Model C- (Proposed)		**65.234%**	**76.516%**

بیشتر بخوانیم

- https://github.com/Alireza-Akhavan/deep-face-recognition
- https://www.aparat.com/v/6Rnxl
- https://www.aparat.com/v/IHlF7

http://iran-celeb.ir/



منابع

- *https://www.coursera.org/specializations/deep-learning*
- *http://blog.class.vision/winter-96-97-syllabus/*
- http://www.highload.ru/2017/abstracts/3044.html
- https://medium.com/@ahmdtaha/facenet-a-unified-embedding-for-face-recognition-and-clustering-7d34abde9
- http://ydwen.github.io/papers/WenECCV16-poster.pdf
- https://arxiv.org/abs/1503.03832
- https://ydwen.github.io/papers/WenECCV16.pdf
- https://arxiv.org/abs/1704.08063
- https://arxiv.org/abs/1801.05599
- https://arxiv.org/abs/1801.07698