Angular triangle distance for ordinal metric learning

Imam Mustafa Kamal & Hyerim Bae*

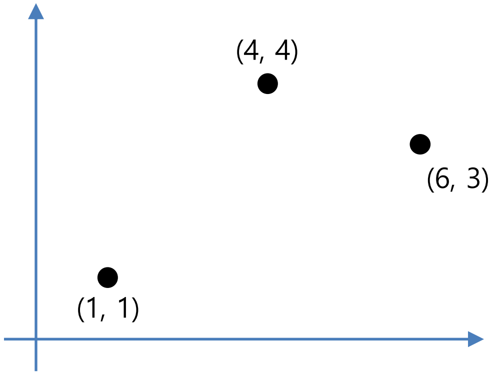
Available at https://arxiv.org/abs/2211.15200

Outline

- Introduction
- Related works
- Problem
- Proposed method
- Experimental results
- Conclusion

Motivation

- How similar?
  - Scalar: 

  - Vector: 

Metric

Euclidean (2^{nd} Norm) $\sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$

P-th Norm $\sqrt[p]{\sum_{i=1}^{n}(x_i - y_i)^p}$

Inner Product $x \cdot y = \sum_{i=1}^{n} x_i y_i$

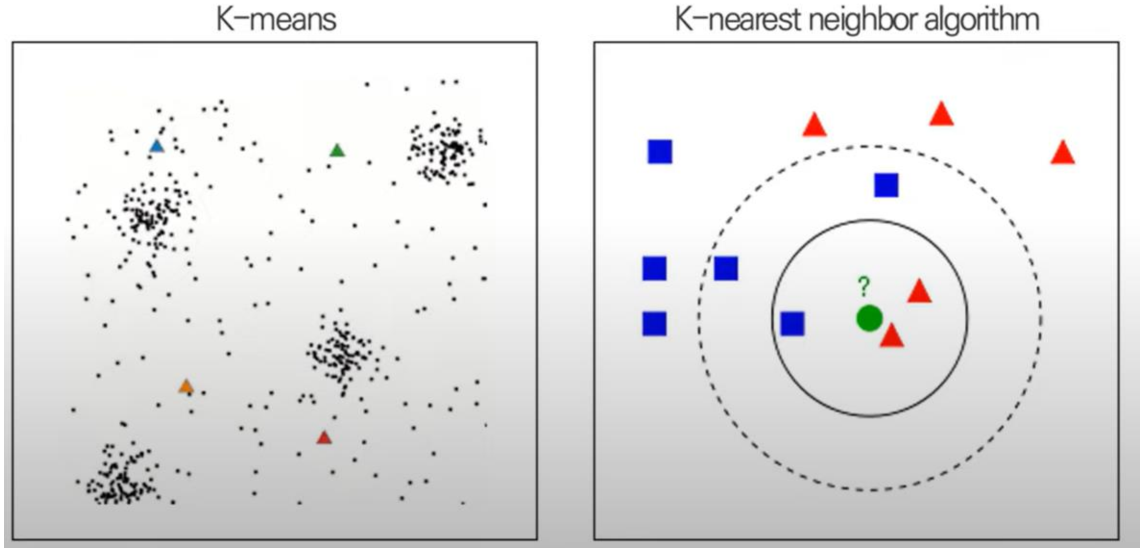
Cosine Distance $1 - \frac{x \cdot y}{\|x\|\|y\|}$

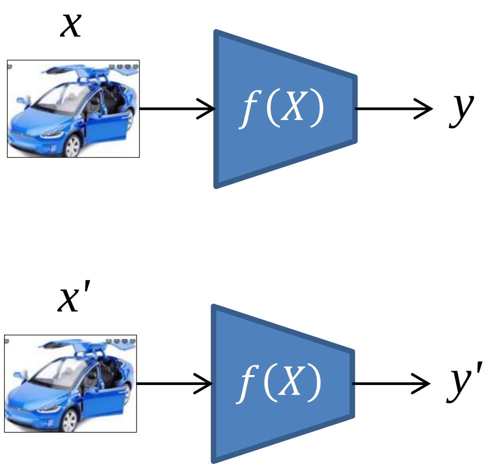
Mahalanobis Distance $\sqrt{(x-y)^T \Sigma^{-1} (x-y)}$

Properties of distance measure

1. Nonnegative $d_{ij} \geq 0$
2. Self-Proximity $d_{ii} = 0$
3. Symmetry $d_{ij} = d_{ji}$
4. Triangular Inequality $d_{ij} \leq d_{ik} + d_{kj}$

- Metric used in Supervised learning and Unsupervised Learning



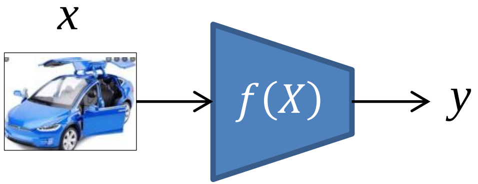


$$L(\theta, x, y)$$

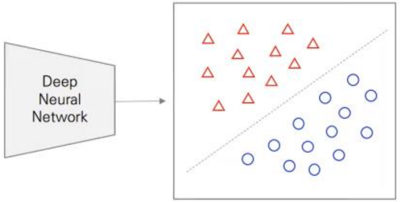
Same or Not?

Motivation

- Classification problem using softmax



$$L(\theta, x, y)$$



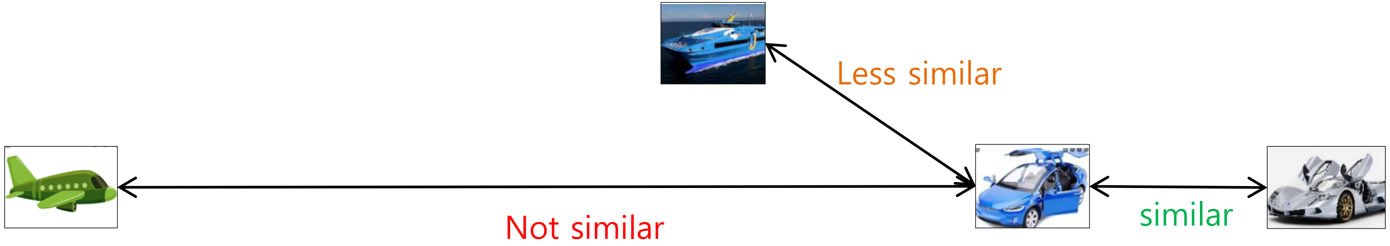
  - Only separable features are used
  - Poor at Openset data

Inter class variation ↑

Intra class variation ↓

Introduction

- How to define a similarity on complex data?

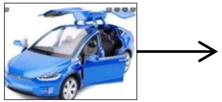


- Metric learning aims to automatically construct task-specific distance or similarity of data yielding low-dimensional representation.

- $Z = f(X),\ X \in \mathbb{R}^P, Z \in \mathbb{R}^q, p \gg q$

Why metric learning (ML)?

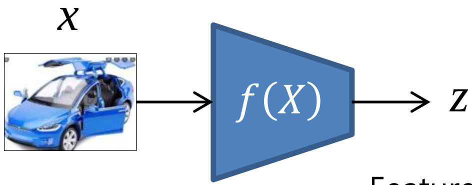
- By using metric learning (ML)
  - ML as a dimensionality reduction
  - ML as a feature extraction
  - ML for fine-tuning model
  - ML for transfer learning



Features = 64x64x3 = 12,288
#samples = 10,000
Minimum #operation = 122,880,000

$X \in \mathbb{R}^P, Z \in \mathbb{R}^q, p \gg q,$

$\mathrm{E.g.}\ p = 12{,}288, q = 50$



Features = 50
#samples = 10,000
Minimum #operation = 500,000

ML can help traditional M/L methods for dealing with complex data

- Information retrieval
- k-NN classification
- Clustering
- Classification

Related works(1)

- Metric learning emerged in 2002 with the pioneering work of Xing et al. (2002)
- $d_M(x, x') = \sqrt{(x - x')^T M (x - x')}$

- Must-link / cannot-link constraints (sometimes called positive / negative pairs):

$$\begin{aligned} \mathcal{S} &= \{(x_i, x_j) : x_i \text{ and } x_j \text{ should be similar}\}, \\ \mathcal{D} &= \{(x_i, x_j) : x_i \text{ and } x_j \text{ should be dissimilar}\}. \end{aligned}$$

- Relative constraints (sometimes called training triplets):

$$\mathcal{R} = \{(x_i, x_j, x_k) : x_i \text{ should be more similar to } x_j \text{ than to } x_k\}.$$

$$\min_{\boldsymbol{M}} \underbrace{\ell(\boldsymbol{M}, \mathcal{S}, \mathcal{D}, \mathcal{R})}_{\textit{Loss}} + \underbrace{\lambda R(\boldsymbol{M})}_{\textit{Regularization}}$$

Disadvantages?

Requires Expensive computation!

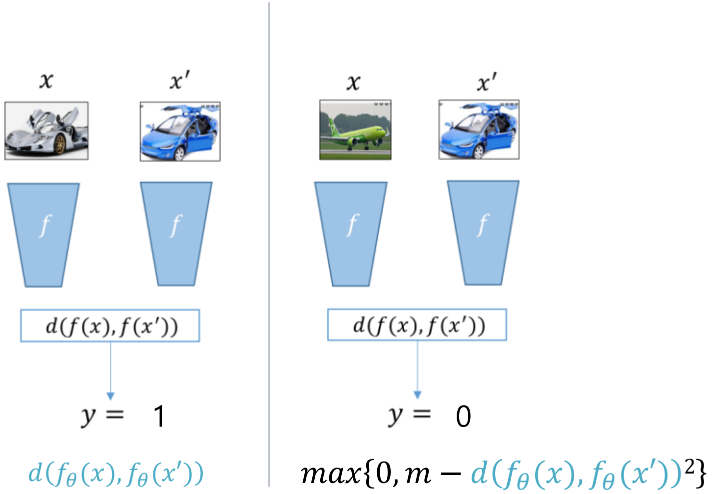
1. An image = 64x64x3 = 12,288
2. Large number of constraints!

Paper: Bellet, Aurélien, Amaury Habrard and Marc Sebban. "A Survey on Metric Learning for Feature Vectors and Structured Data." *ArXiv* abs/1306.6709 (2013).

Related works(2)

- Metric learning with (deep) neural network
- Contrastive loss

$$l_{cont.} = y(d(f_\theta(x), f_\theta(x'))^2) \;\; + (1-y)(max\{0, m - d(f_\theta(x), f_\theta(x'))^2\})$$



If the two are similar (y=1), minimize the distance
If the two are different (y=0),
  if the distance exceeds m (already large), do nothing
  if the distance is smaller than m, enlarge the distance

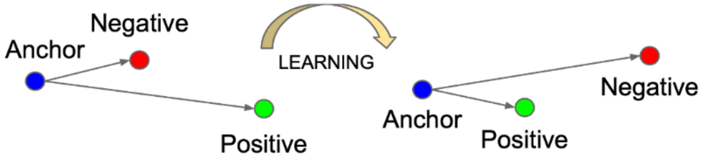
Disadvantages?

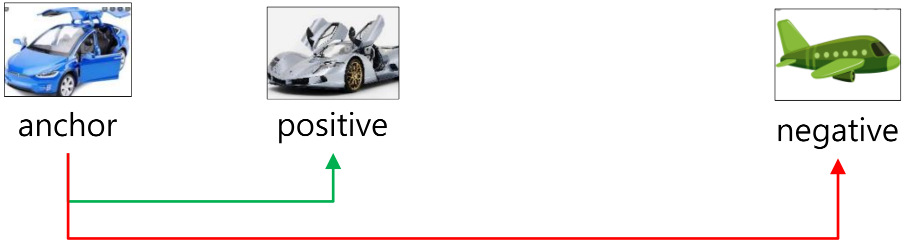
- It cannot provide relative constraints

Paper: S. Chopra, R. Hadsell and Y. LeCun, "Learning a similarity metric discriminatively, with application to face verification," *IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (2005).

Related works(3)

- Triplet loss





$$l_{triplet} = \max(\underbrace{||A-P||^2}_{\left(d(f_\theta(x_a), f_\theta(x_p))\right)^2} - \underbrace{||A-N||^2}_{\left(d(f_\theta(x_a), f_\theta(x_n))\right)^2} + m, 0)$$

?

Disadvantages?

- Hard to converge
- High computation

Paper: Florian Schroff, Dmitry Kalenichenko, and James Philbin. "Facenet: A unified embedding for face recognition and clustering." *IEEE Conference on Computer Vision and Pattern Recognition* (2015).

Related works(4)

- Quadruplet loss

anchor

positive

negative 1

negative 2

$$l_{quad.} = \{||A-P||^2 - ||A-N1||^2 + m1\} + \{||A-P||^2 - ||A-N2||^2 + m2\}$$

Disadvantages?

- High computation

Paper: Weihua Chen, Xiaotang Chen, Jianguo Zhang, and Kaiqi Huang."Beyond triplet loss: A deep quadruplet network for person re-identification." *IEEE Conference on Computer Vision and Pattern Recognition* (2017).

Related works(5)

- N-pair loss

Given a $(N+1)$-tuplet of training samples, $\{\mathbf{x}, \mathbf{x}^+, \mathbf{x}_1^-, \ldots, \mathbf{x}_{N-1}^-\}$, including one positive and $N-1$ negative ones, N-pair loss is defined as:

$$\mathcal{L}_{\text{N-pair}}(\mathbf{x}, \mathbf{x}^+, \{\mathbf{x}_i^-\}_{i=1}^{N-1}) = \log\left(1 + \sum_{i=1}^{N-1} \exp(f(\mathbf{x})^\top f(\mathbf{x}_i^-) - f(\mathbf{x})^\top f(\mathbf{x}^+))\right)$$
$$= -\log \frac{\exp(f(\mathbf{x})^\top f(\mathbf{x}^+))}{\exp(f(\mathbf{x})^\top f(\mathbf{x}^+)) + \sum_{i=1}^{N-1} \exp(f(\mathbf{x})^\top f(\mathbf{x}_i^-))}$$

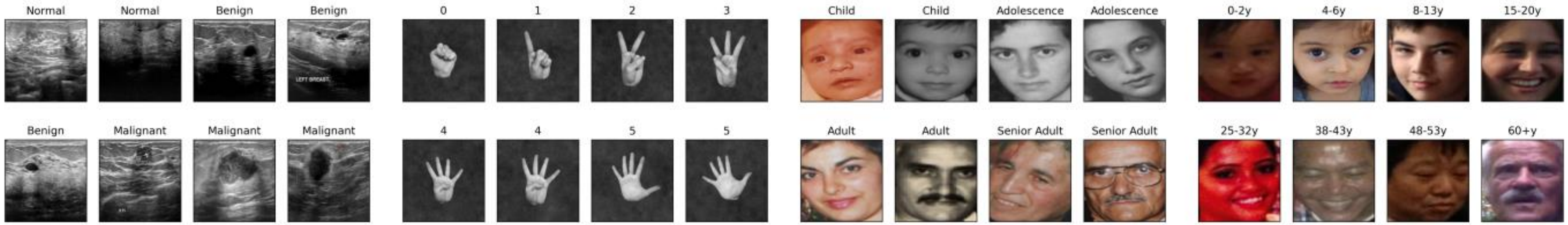
Disadvantages?

- High computation
- Requires large N to obtain a proper results

Paper: Sohn Kihyuk. "Improved Deep Metric Learning with Multi-class N-pair Loss Objective." *Neurips* (2016).

Problem

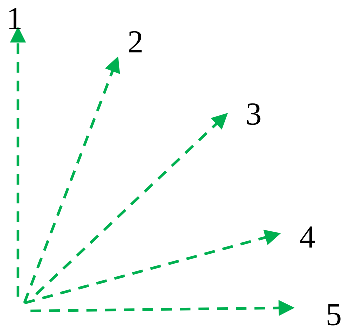
- None deep metric learning methods guarantee to preserve the ordinal nature of original data in low-dimensional space.
- Data with ordinal nature is ubiquitous in real-world problems

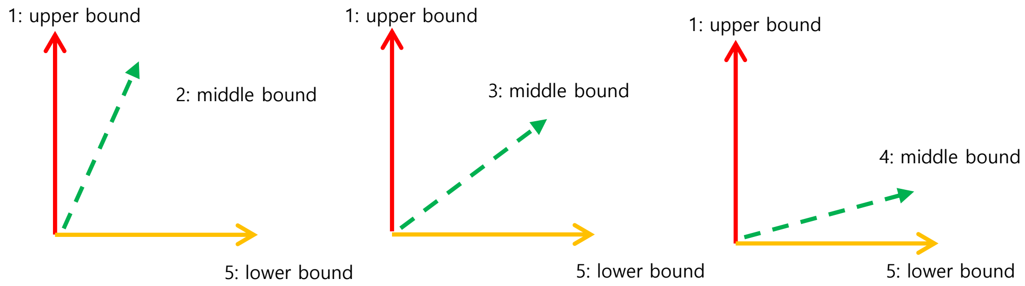


(a) Busi (b) Finger (c) FG-Net (d) Adience

Angular distance

- In order to learn the ordinal relations,





Learning every pair of combinations:
11,12,13,14,15,22,23,24,25,33,34,35,44,45,55

Learning every triple of combinations:
111,222,333,444,555,125,135,145,151

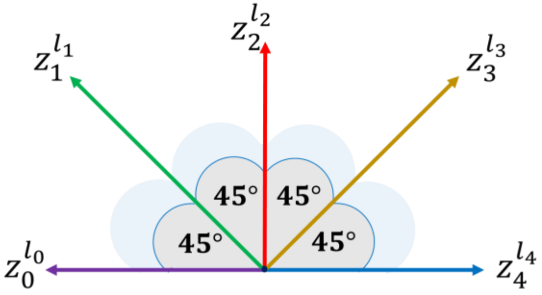
$$C + \binom{C}{2} \longrightarrow C+(C-1)$$

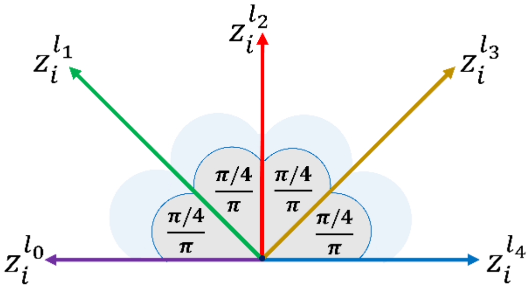
Proposed method

- Suppose there are five number of (ordinal) class: l_0 , l_1 , l_2 , l_3 , and l_4
- D_A = Angular distance

$$D_A(z_i^{l_{r_i}}, z_j^{l_{r_j}}) = \frac{\cos^{-1}(S_C(z_i^{l_{r_i}}, z_j^{l_{r_j}}))}{\pi} = \frac{\theta_{z_i^{l_{r_i}}, z_j^{l_{r_j}}}}{\pi}$$

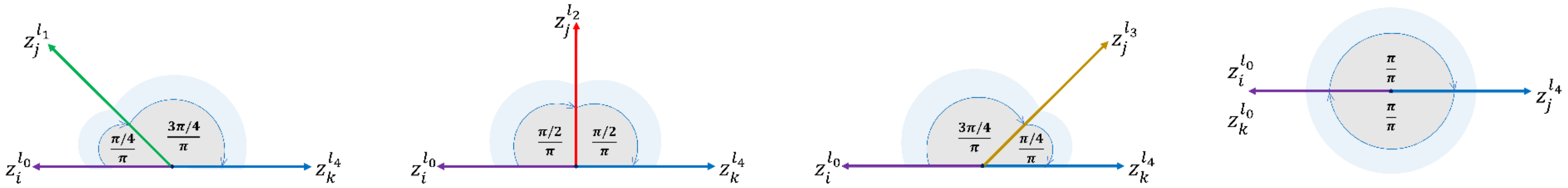
- Sc = Cosine similarity



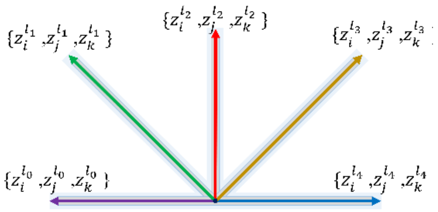


Proposed method

- Learn the distance between lower-bond to middle-bond and middle-bound to upper-bond

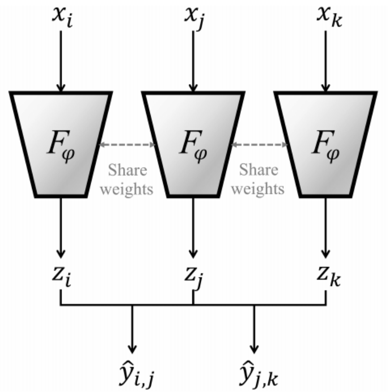


- Learn the inner-class distances



Proposed method

- Ordinal triplet network



$$y_{i,j} = D_A(x_i, x_j),\ y_{j,k} = D_A(x_j, x_k)$$

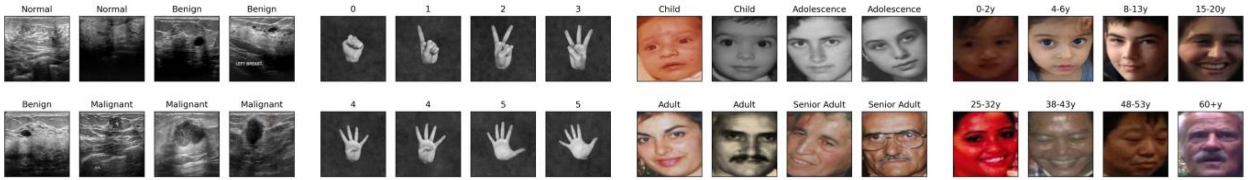
$$\hat{y}_{i,j} = \widehat{D}_A(x_i, x_j),\ \hat{y}_{j,k} = \widehat{D}_A(x_j, x_k)$$

Algorithm 1: training procedure of ordinal triplet network

Input: $\mathcal{X} = \{x_i, x_j, x_k\}_0, \ldots, \{x_i, x_j, x_k\}_{T-1}, \mathcal{Y} = \{y_{ij}, y_{jk}\}_0, \ldots, \{y_{ij}, y_{jk}\}_{T-1}$
Output: F_φ^*
for $e \leftarrow 0$ **to** $Epoch - 1$ **do**
  $z_i = F_{\varphi_e}(x_i), z_j = F_{\varphi_e}(x_j), z_k = F_{\varphi_e}(x_k)$ `// extract embedding reprsentation`
  $\hat{y}_{ij} = D_A(z_i, z_j), \hat{y}_{jk} = D_A(z_j, z_k)$ `//` D_A `is denoted in Eq. 1`
  $\mathcal{L}_{\varphi_e} = \mathrm{MSE}(y_{i,j}, \hat{y}_{i,j}) + \mathrm{MSE}(y_{j,k}, \hat{y}_{j,k})$ `// calculate loss (Eq. 5)`
  $\varphi_e \leftarrow \varphi_e - \eta \nabla_{\varphi_e} \mathcal{L}_{\varphi_e}$ `// update model parameter`
end

How are intra- and inter-class learning possible?

Results(1)



(a) Busi (b) Finger (c) FG-Net (d) Adience

Table 1: Evaluating the embedding space separability with an SVM classifier.

Model	Accuracy (mean ± std.)			
	Busi	Finger	FG-Net	Adience
O-Net	**0.950** ± 0.026	**1.000** ± 0.000	**0.961** ± 0.010	**0.578** ± 0.153
S-Net	0.903 ± 0.040	**1.000** ± 0.000	0.957 ± 0.015	0.547 ± 0.161
T-Net	0.564 ± 0.023	**1.000** ± 0.000	0.864 ± 0.029	**0.578** ± 0.178
Q-Net	0.879 ± 0.027	**1.000** ± 0.000	0.905 ± 0.016	0.545 ± 0.051
N-Net_1	0.558 ± 0.009	**1.000** ± 0.000	0.961 ± 0.026	0.570 ± 0.159
N-Net_2	0.923 ± 0.024	**1.000** ± 0.000	0.919 ± 0.008	0.458 ± 0.157

All methods performs perfectly in the Finger dataset

Results(2)

- Test in ordinal data set

Table 4: K-NN classification error on the datasets with ordinal features. $K = 3$

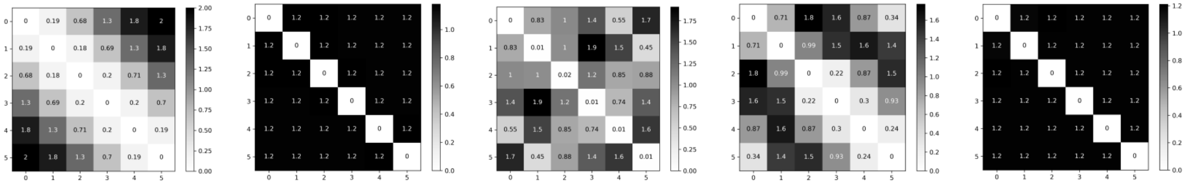
Model	Error rate (mean ± std.)			
	Car	Nursery	Hayes-Roth	Balance
Real-Eucl [8]	11.4 ± 0.7	8.6 ± 0.1	38.5 ± 3.1	15.2 ± 1.1
Real-LMNN [8]	5.0 ± 0.3	2.4 ± 0.1	23.1 ± 1.6	17.6 ± 0.9
Binary-Eucl [8]	24.0 ± 1.4	24.0 ± 0.2	50.0 ± 2.9	32.7 ± 1.9
Binary-LMNN [8]	4.1 ± 0.3	2.3 ± 0.1	$\mathbf{16.0} \pm 1.0$	17.8 ± 1.2
Binary-Ord-Eucl [8]	12.3 ± 0.4	8.7 ± 0.1	45.5 ± 3.3	16.7 ± 0.8
Binary-Ord-LMNN [8]	4.1 ± 0.2	1.9 ± 0.1	15.4 ± 1.2	13.4 ± 0.8
Ex-Gower [8]	12.1 ± 0.7	8.8 ± 0.1	37.2 ± 1.9	32.8 ± 1.3
Thresh-Eucl [8]	4.5 ± 0.4	2.3 ± 0.1	22.3 ± 1.9	14.5 ± 0.7
Ord-LMNN-Uni [8]	3.8 ± 0.3	1.8 ± 0.1	20.5 ± 1.3	6.8 ± 0.5
Ord-LMNN-Beta [8]	3.7 ± 0.3	1.6 ± 0.1	18.6 ± 1.0	$\mathbf{6.1} \pm 0.5$
Ord-LMNN-RecBeta [8]	3.4 ± 0.3	1.6 ± 0.1	18.6 ± 1.0	6.4 ± 0.5
O-Net	$\mathbf{3.1} \pm 0.4$	$\mathbf{0.8} \pm 0.1$	$\mathbf{16.0} \pm 1.5$	$\mathbf{6.1} \pm 0.6$

- ❑ Our method outperforms traditional ordinal metric learning method

[8] Yuan Shi, Wenzhe Li, and Fei Sha. Metric learning for ordinal data. *Proceedings of the AAAI Conference on Artificial Intelligence*, 30(1), Mar. 2016. doi: 10.1609/aaai.v30i1.10280.

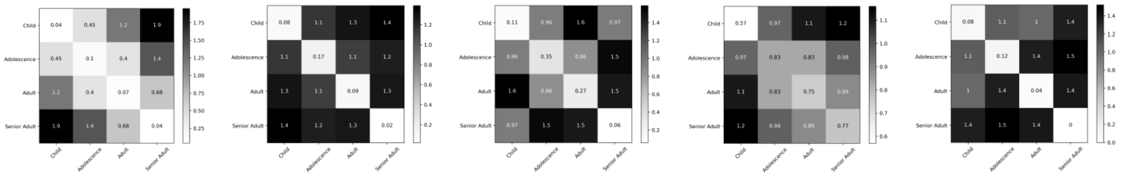
Results(3)

- Only our method can preserve the ordinal nature of data in a low-dimensional space representation



(a) O-Net (b) S-Net (c) T-Net (d) Q-Net (e) N-Net$_2$

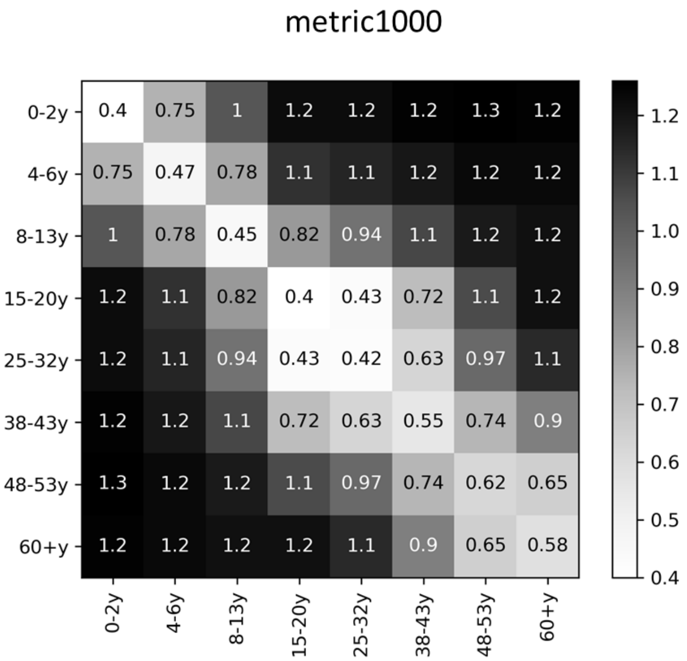
Figure 6: Pairwise cosine distance matrix of latent representation in the Finger dataset.



(a) O-Net (b) S-Net (c) T-Net (d) Q-Net (e) N-Net$_2$

Figure 7: Pairwise cosine distance matrix of latent representation in the FG-Net dataset.

- Audience face dataset



Conclusion

- Ordinal data are widespread in real-world problems, but they have been frequently considered standard nominal problems, which can lead to non-optimal solutions.
- This study introduces a new deep metric learning (DML) method for solving ordinal data
- Our method obtained more accurate and semantic embedding space representation compared with the existing Metric learning models.

- Limitation: as a triplet representation, it potentially requires a high computational cost in a large-scale dataset.
- This issue leads to another research direction to be addressed in the future

References

[1]. Bellet, Aurélien, Amaury Habrard and Marc Sebban. "A Survey on Metric Learning for Feature Vectors and Structured Data." ArXiv abs/1306.6709 (2013).

[2]. Jane Bromley, Isabelle Guyon, Yann LeCun, Eduard Säckinger, and Roopak Shah. "Signature verification using a 'siamese' time delay neural network." Advances in Neural Information Processing Systems (1993).

[3]. S. Chopra, R. Hadsell and Y. LeCun, "Learning a similarity metric discriminatively, with application to face verification," IEEE Computer Society Conference on Computer Vision and Pattern Recognition (2005).

[4]. Florian Schroff, Dmitry Kalenichenko, and James Philbin. "Facenet: A unified embedding for face recognition and clustering." IEEE Conference on Computer Vision and Pattern Recognition (2015).

[5]. Weihua Chen, Xiaotang Chen, Jianguo Zhang, and Kaiqi Huang."Beyond triplet loss: A deep quadruplet network for person re-identification." IEEE Conference on Computer Vision and Pattern Recognition (2017).

[6]. Sohn Kihyuk. "Improved Deep Metric Learning with Multi-class N-pair Loss Objective." Neurips (2016).

[7]. IM Kamal, H. Bae, L. Liu. "Metric Learning as a Service with Covariance Embedding." IEEE Transactions on service computing (under review).

[8]. IM Kamal, H. Bae. "Metric Learning as a Service with Covariance Embedding." IEEE Transactions on service computing (under review).