18th Iberoamerican Congress on Pattern Recognition, Havana CUBA

Auto-encoder Based Data Clustering

Chunfeng Song[1], Feng Liu[2], Yongzhen Huang[1], Liang Wang[1] and Tieniu Tan[1]

1. NLPR, Chinese Academy of Sciences
2. Southeast University

CIARP 2013
18th Iberoamerican Congress on Pattern Recognition, Havana CUBA (20-23 Nov)

●Data clustering

Previous clustering method

- ✓ K-means
- ✓ Spectral clustering
- ✓ N-cut

→ Linear mapping → Perform bad for bad distributed data.

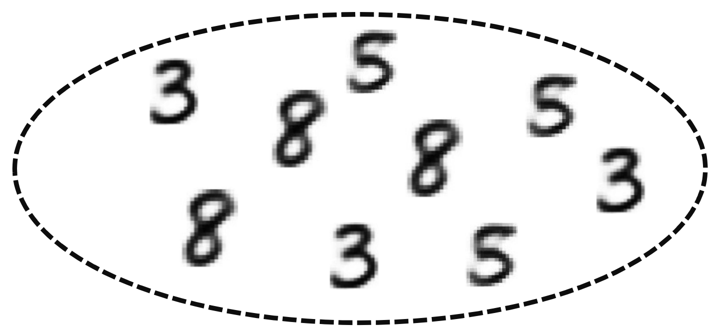
Data clustering

Previous clustering method

- ✓ K-means
- ✓ Spectral clustering
- ✓ N-cut

→ Linear mapping → Perform bad for bad distributed data.

Can not deal with this similar images



Original Data Space

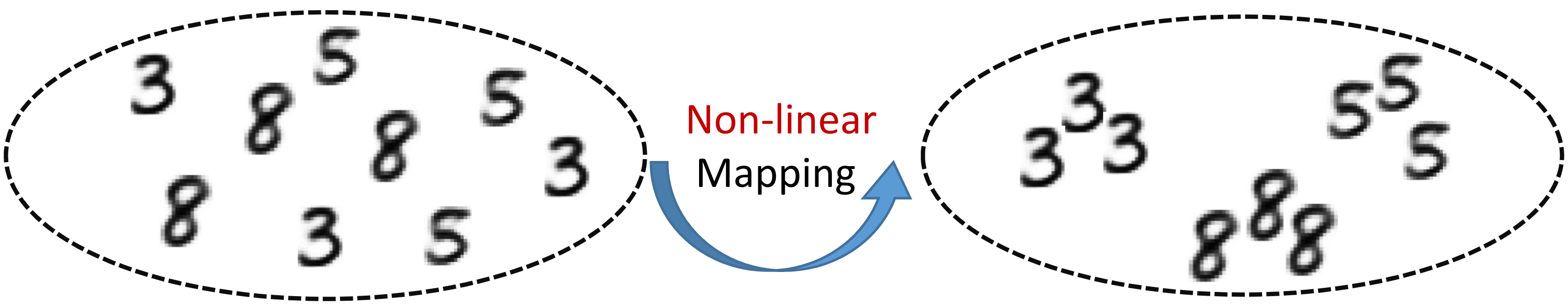
● Data clustering

Previous clustering method

- ✓ K-means
- ✓ Spectral clustering
- ✓ N-cut

→ Linear mapping → Perform bad for bad distributed data.

✓ Auto-encoder based clustering can provide non-linear mapping.

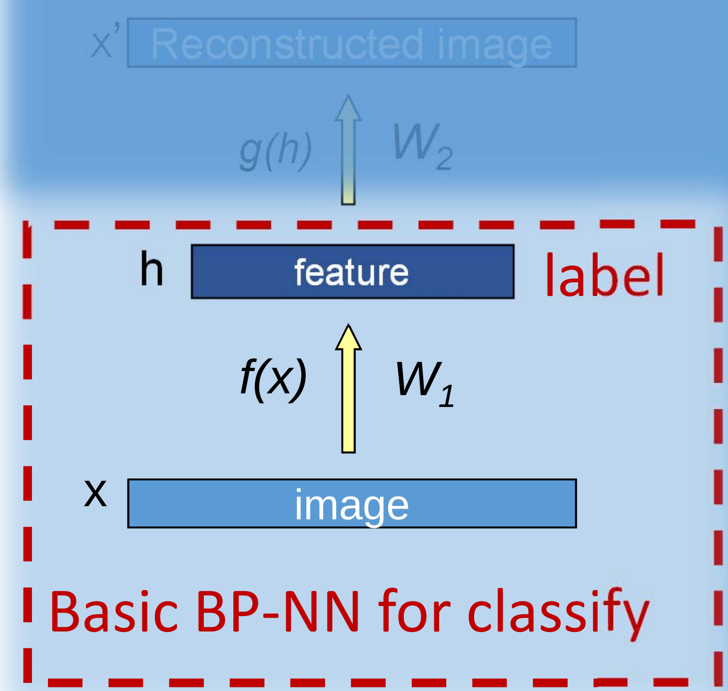


Original Data Space

Feature Space

Auto-encoder

Basic single-layer auto-encoder → Is a kind of BP-NN



supervised

Encoder function

Sigmiod-type

$$h_i = f(x_i) = \frac{1}{1+\exp(-(W_1 x_i + b_1))}$$

Decoder function

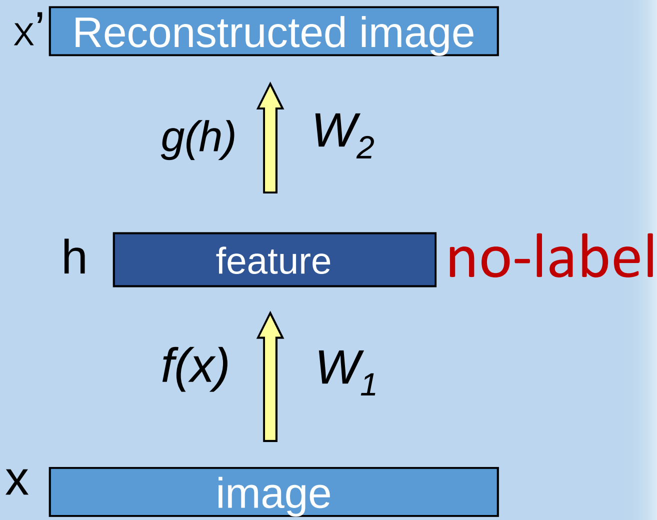
$$x'_i = g(h_i) = \frac{1}{1+\exp(-(W_2 h_i + b_2))}$$

Obj. function

$$\min \frac{1}{N}\sum_{i=1}^{N}\left\| x_i - x_i' \right\|^2$$

Auto-encoder

Basic single-layer auto-encoder → Is a kind of BP-NN



Encoder function

Sigmiod-type

$$h_i = f(x_i) = \frac{1}{1+\exp(-(W_1 x_i + b_1))}$$

Decoder function

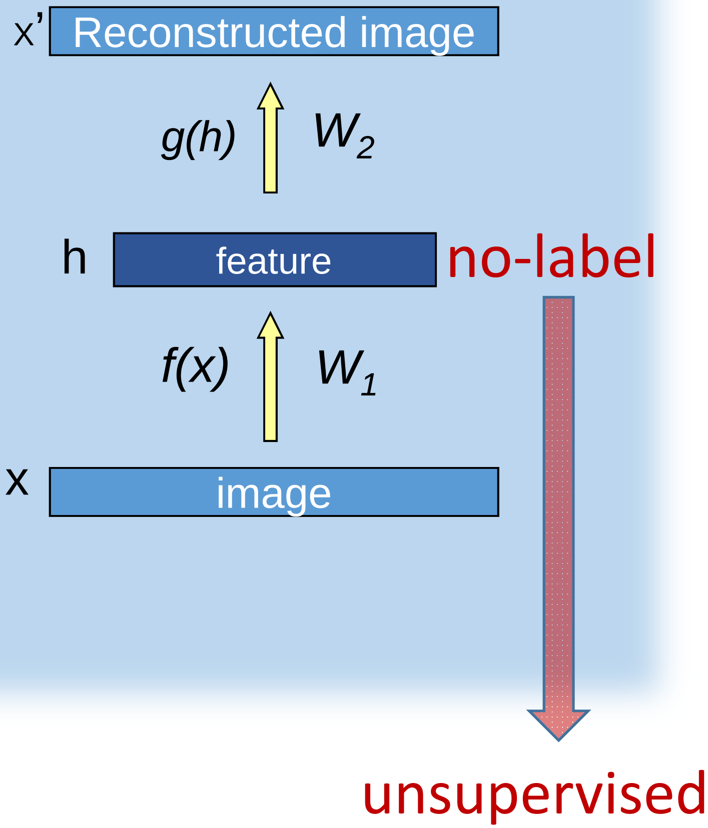
$$x'_i = g(h_i) = \frac{1}{1+\exp(-(W_2 h_i + b_2))}$$

Obj. function

$$\min \frac{1}{N}\sum_{i=1}^{N} \left\| x_i - x_i' \right\|^2$$

Auto-encoder

Basic single-layer auto-encoder → Is a kind of BP-NN



Encoder function

Sigmiod-type

$$h_i = f(x_i) = \frac{1}{1+\exp(-(W_1 x_i + b_1))}$$

Decoder function

$$x'_i = g(h_i) = \frac{1}{1+\exp(-(W_2 h_i + b_2))}$$

Obj. function

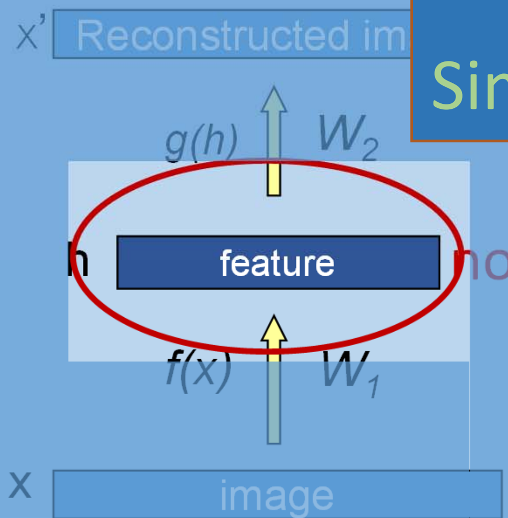
$$\min \frac{1}{N} \sum_{i=1}^{N} \left\| x_i - x_i' \right\|^2$$

Auto-encoder

Basic single-layer auto-encoder → Is a kind of BP-NN

Different images may have **different** feature,

Similar images have **similar** feature!



no-label

unsupervised

Decoder function

$$x'_i = g(h_i) = \frac{1}{1+\exp(-(W_2 h_i + b_2))}$$

Obj. function

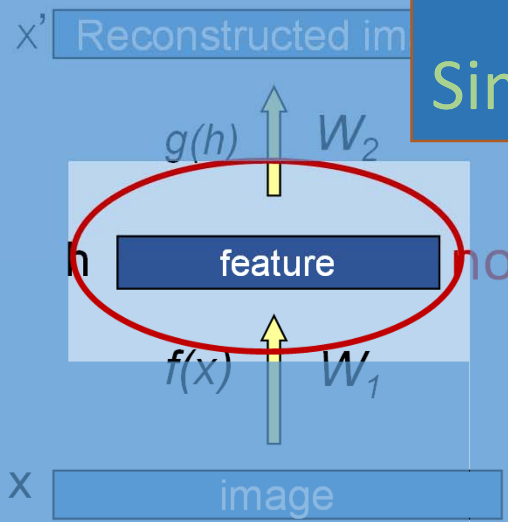
$$\min \frac{1}{N}\sum_{i=1}^{N}\left\| x_i - x_i' \right\|^2$$

Auto-encoder

Basic single-layer auto-encoder → Is a kind of BP-NN

Different images may have different feature,

Similar images have similar feature!



no-label

unsupervised

Decoder function

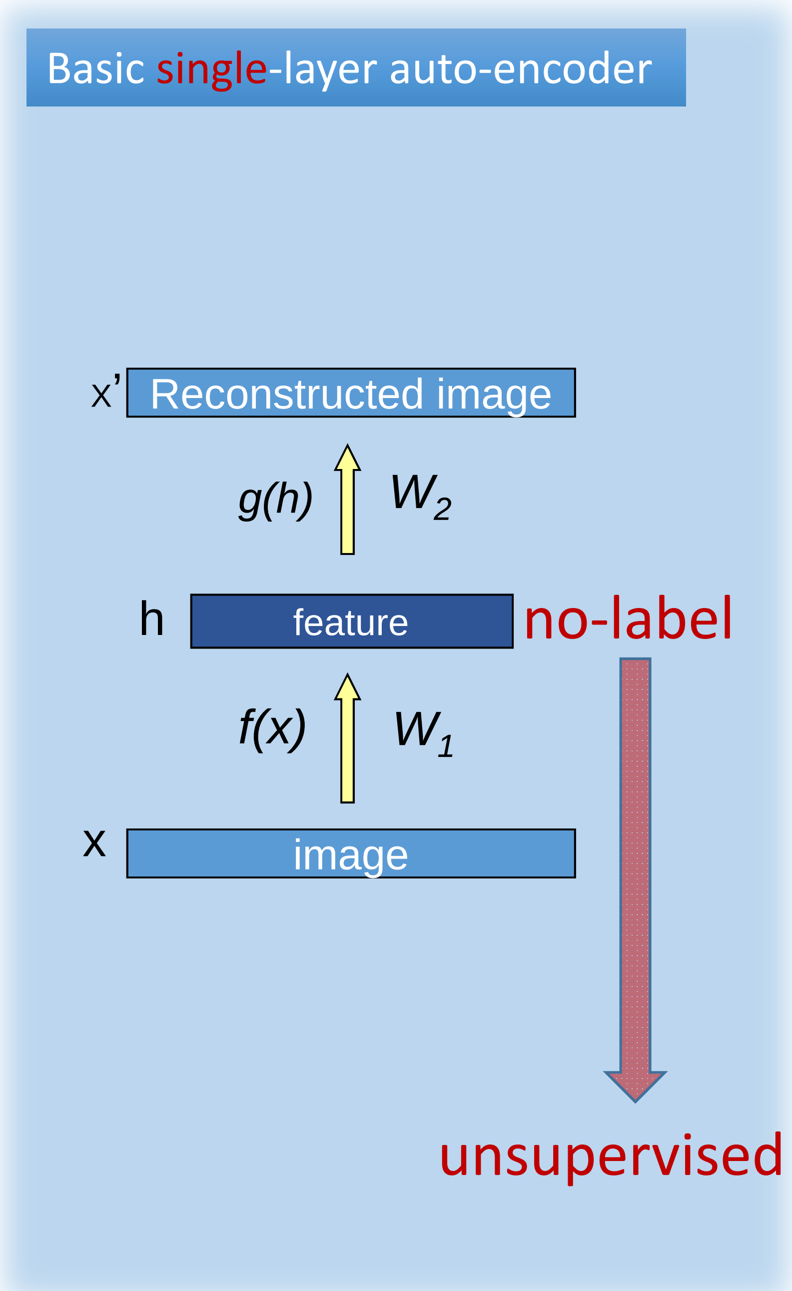
$$x \qquad = \frac{1}{1+\exp(-(W_2 h_i + b_2))}$$

Be fit for clustering

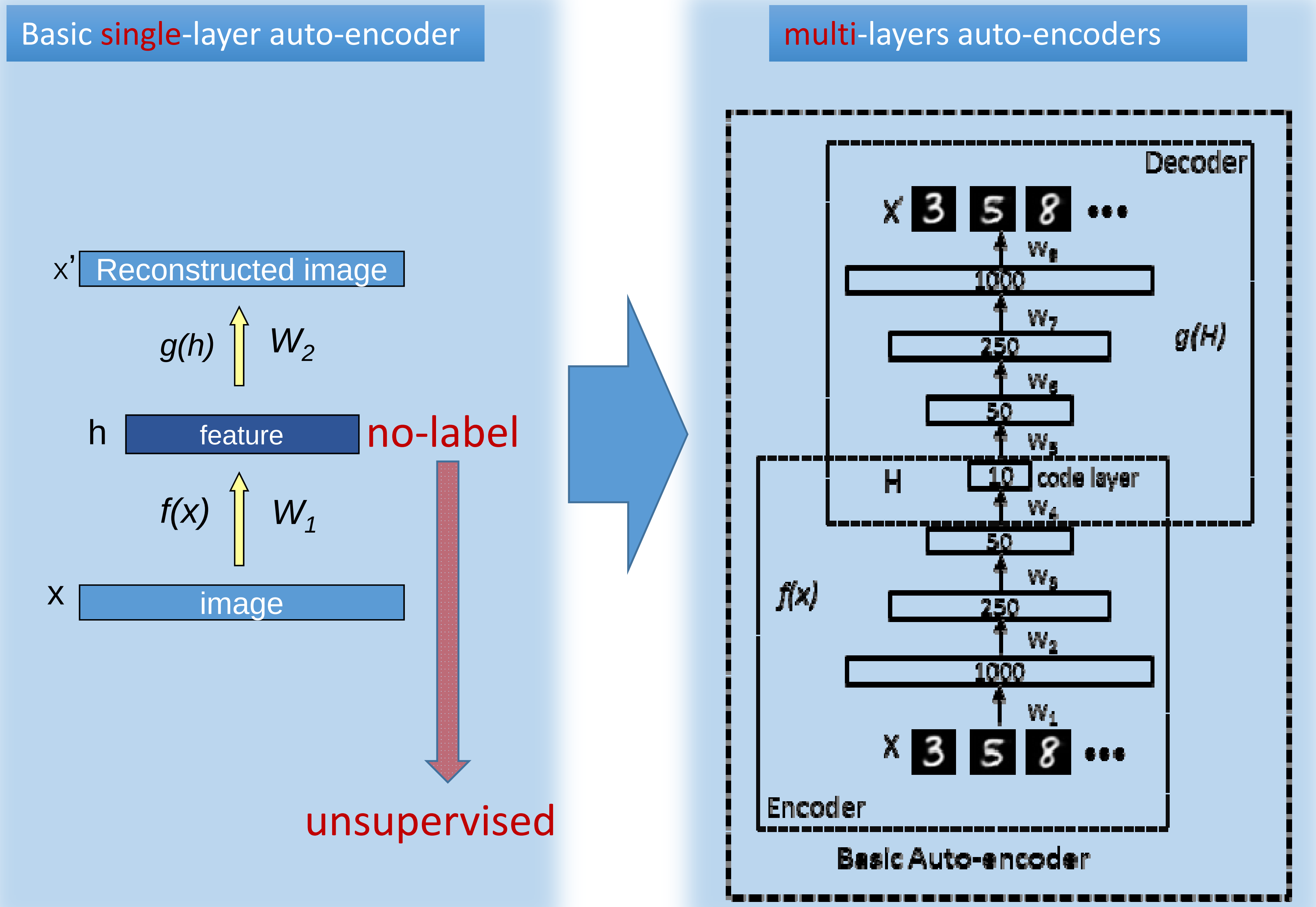
Obj. function

$$\min \frac{1}{N}\sum_{i=1}^{N}\left\| x_i - x_i' \right\|^2$$

Auto-encoder

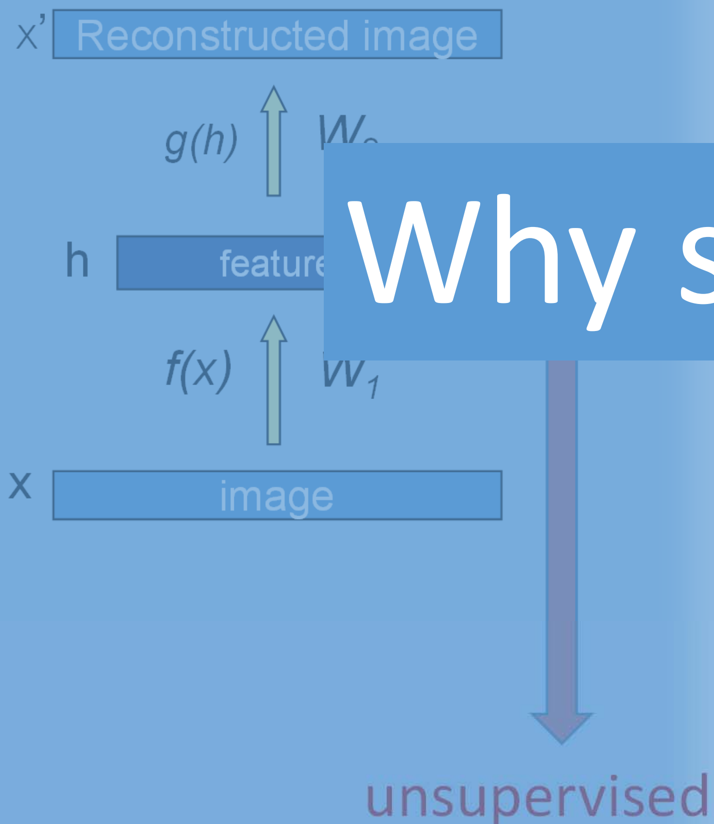


Auto-encoder

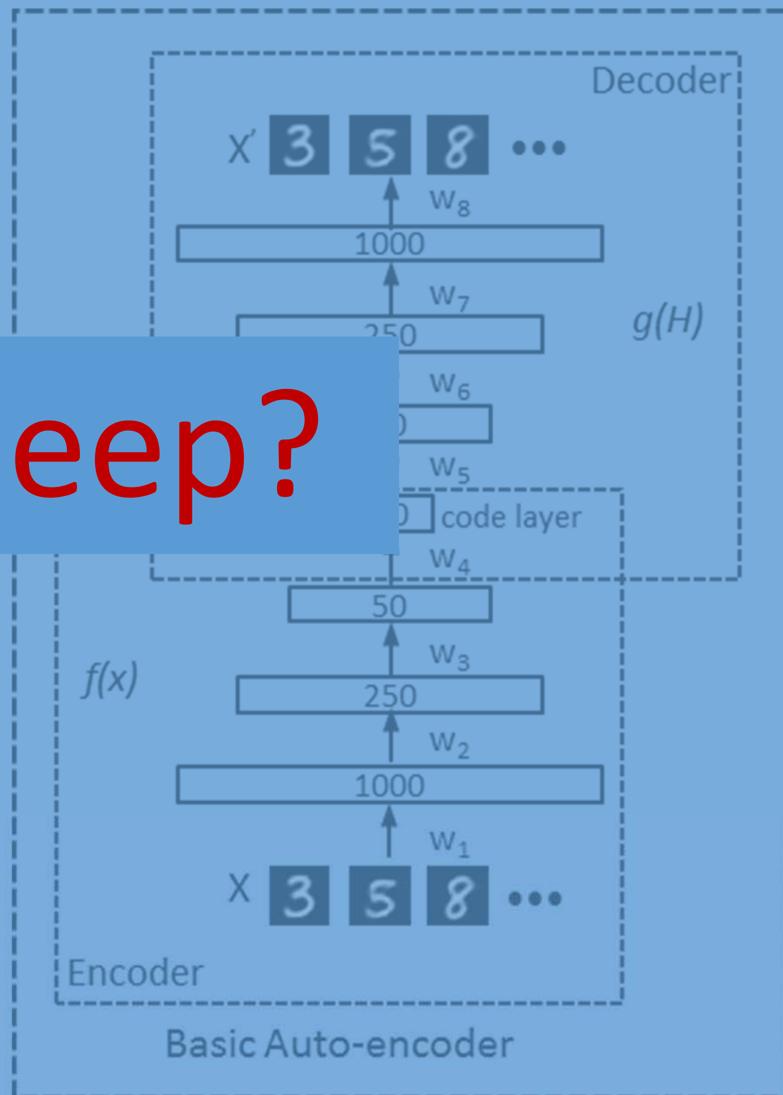


Auto-encoder

Basic single-layer auto-encoder



multi-layers auto-encoders



Why so deep?

Why so deep?

- ✓Deep makes more accurate results.
- ✓Deep networks can learn better.
- ✓Deep networks can provide better non-linear mapping.

Why so deep?

- ✓Deep makes more accurate results
- ✓Deep networks can learn better
- ✓Deep networks can provide better non-linear mapping

Is auto-encoder perfect for clustering?

Why so deep?

✓Deep makes more accurate results

✓Deep networks can learn better

✓Deep networks can provide better non-linear mapping

Is auto-encoder perfect for clustering?

Not enough.

● Clustering Based on Auto-encoder

Obj. fun

Basic auto-encoder

$$\min_{W,b} \frac{1}{N} \sum_{i=1}^{N} \|x_i - x_i'\|^2$$

Clustering Based on Auto-encoder

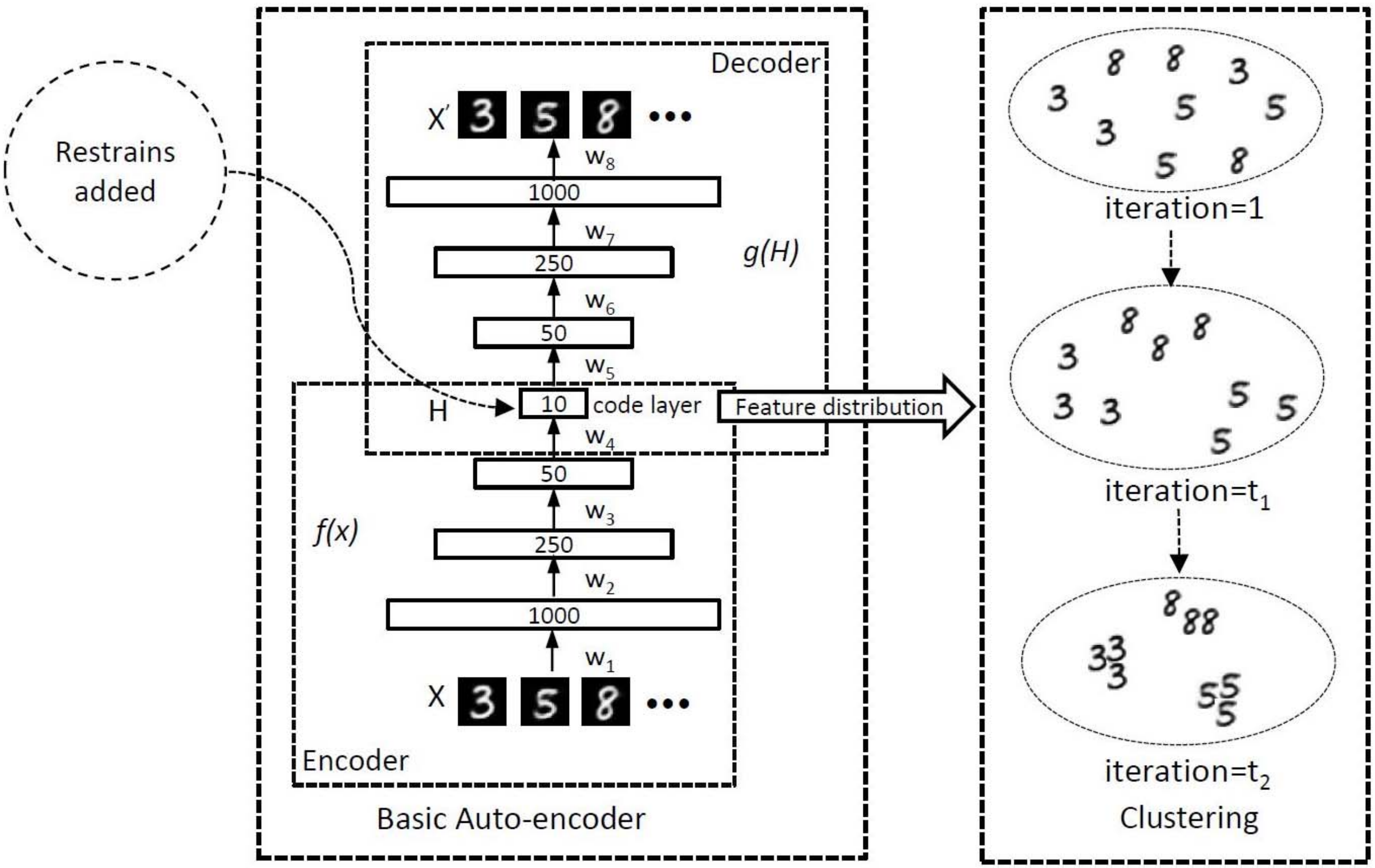
Obj. fun

Basic auto-encoder

$$\min_{W,b} \frac{1}{N}\sum_{i=1}^{N}\|x_i - x_i'\|^2 - \lambda \cdot \sum_{i=1}^{N}\|f^t(x_i) - c_i^*\|^2$$

proposed

✓ Restrains added to achieve compact distribution in feature layer



- Algorithm

$$\min_{W,b} \frac{1}{N} \sum_{i=1}^{N} \|x_i - x_i'\|^2 - \lambda \cdot \sum_{i=1}^{N} \|f^t(x_i) - c_i^*\|^2 \tag{4}$$

$$c_i^* = \arg\min_{c_j^{t-1}} \|f^t(x_i) - c_j^{t-1}\|^2, \tag{5}$$

$$c_j^t = \frac{\sum_{x_i \in C_j^{t-1}} f^t(x_i)}{|C_j^{t-1}|}, \tag{6}$$

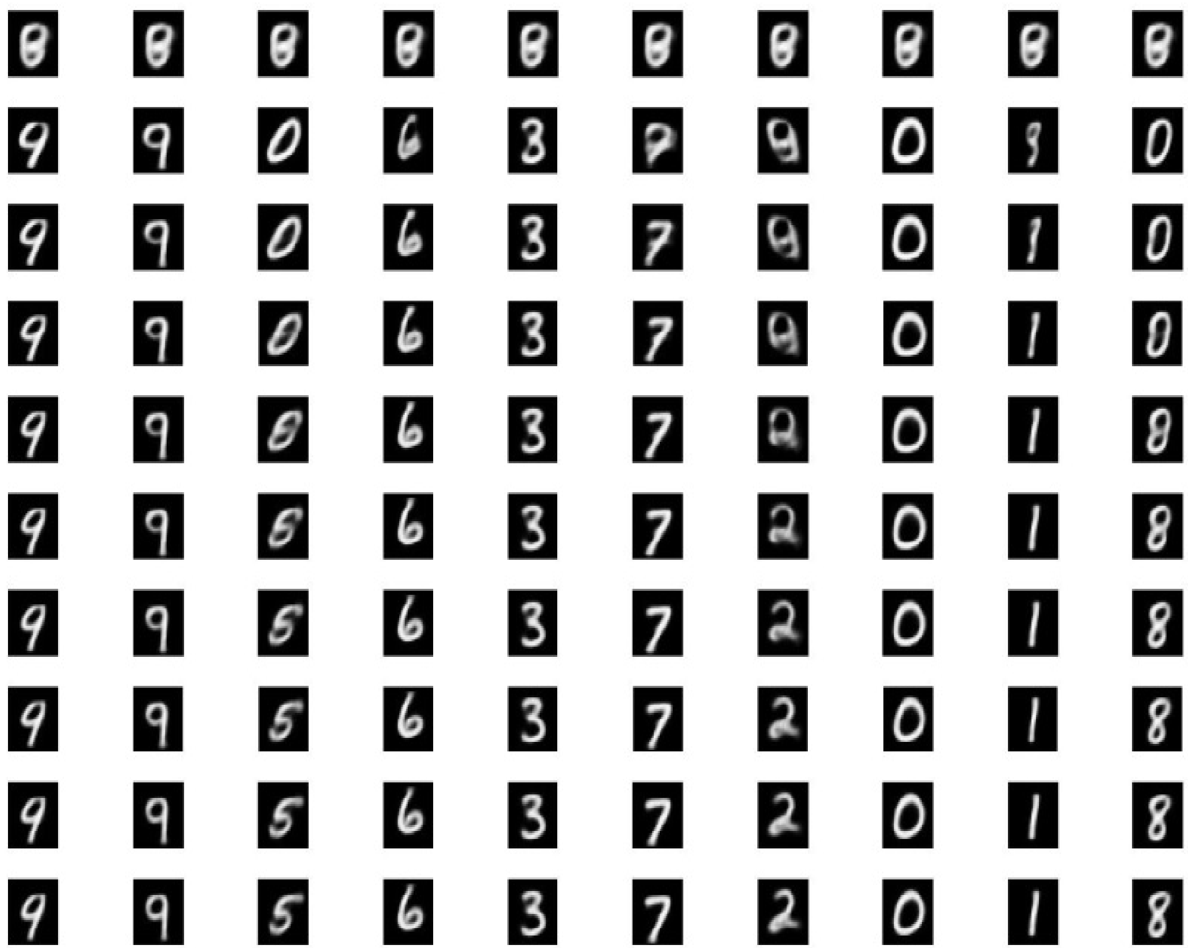
Algorithm 1 Auto-encoder based data clustering algorithm

Input: Dataset X, the number of clusters K, hyper-parameter λ, the maximum number of iterations T.
Initialize sample assignment C^0 randomly.
Set t to 1.
repeat
    Update the mapping network by minimizing Eqn. (4) with stochastic gradient descent for one epoch.
    Update cluster center c^t via Eqn. (6).
    Partition X into K clusters and update the sample assignment C^t via Eqn. (5).
    $t = t + 1$.
until $t > T$
Output: Final sample assignment C.

Iteration

ACC: the cluster accuracy. Distance: the sum of distances between 10 clusters in feature layer.

epoch	ACC	Distance	Visualization of 10 cluster centers(the reconstruction of feature)
1	0.30	0.003	
2	0.46	0.296	
3	0.53	0.432	
4	0.56	0.493	
5	0.59	0.515	
6	0.61	0.526	
7	0.63	0.534	
8	0.65	0.537	
9	0.67	0.538	
10	0.68	0.539	

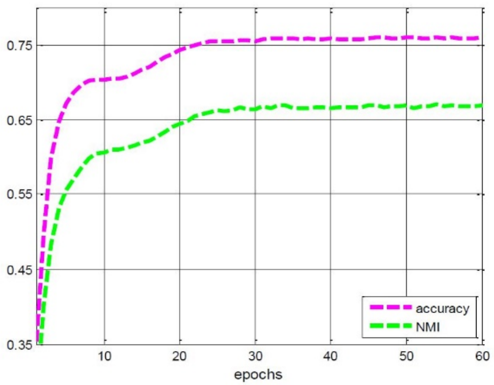


Test on MNIST datasets (including 60000 images with 28*28 resolution)

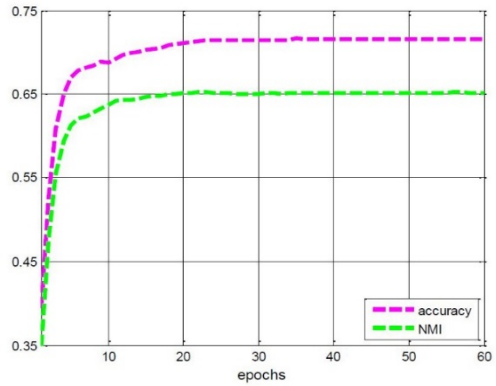
Experiments

✓ Influence of the iteration number on three databases

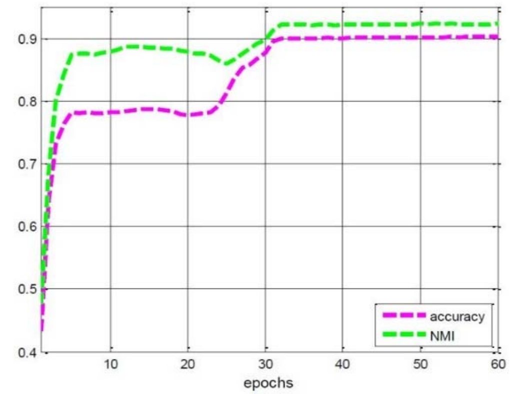
MNIST



USPS



YaleB



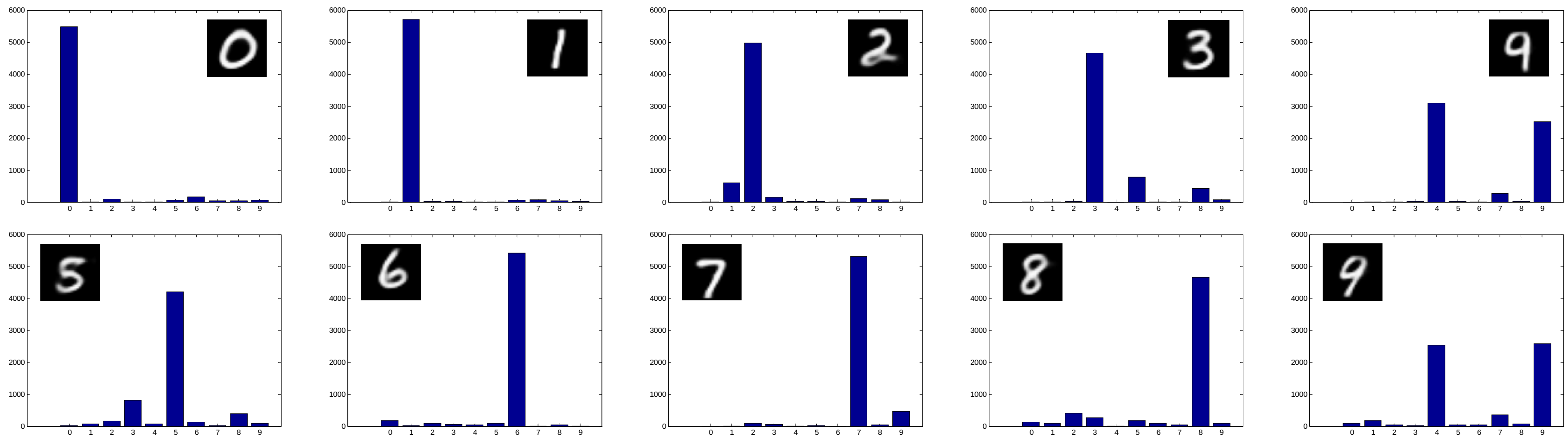
✓ Performance comparison of clustering algorithms on three databases

Datasets	MNIST		USPS		YaleB	
Criterion	NMI	ACC	NMI	ACC	NMI	ACC
K-means	0.494	0.535	0.615	0.674	0.866	0.793
Spectral	0.482	0.556	**0.662**	0.693	0.881	0.851
N-cut	0.507	0.543	0.657	0.696	0.883	0.821
Proposed	**0.669**	**0.760**	0.651	**0.715**	**0.923**	**0.902**

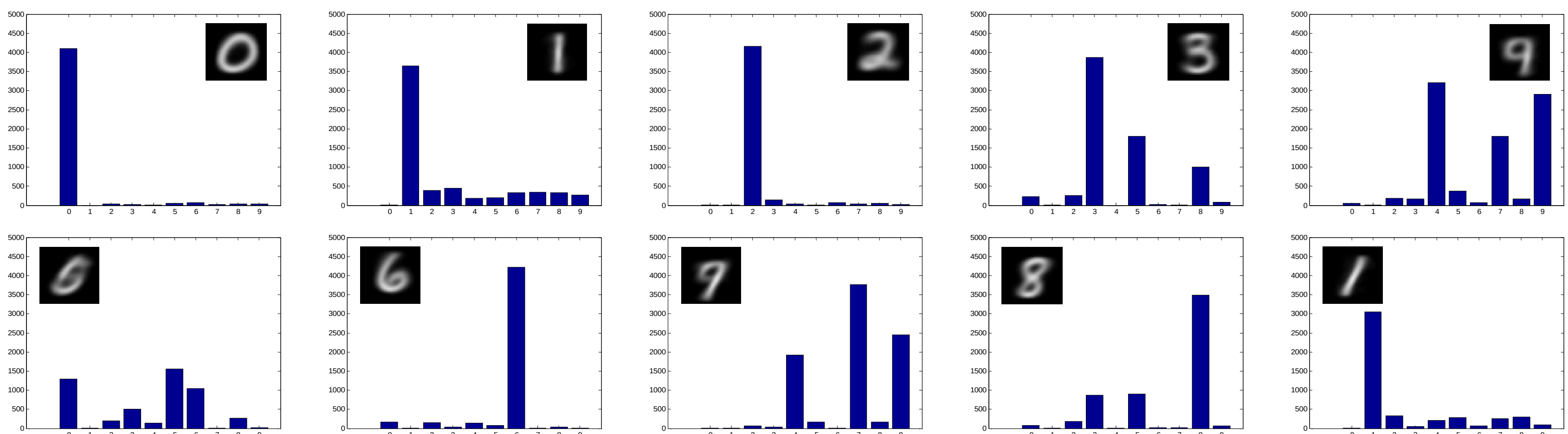
●Experiments

Distribution of data over ten clusters and the visualized images of cluster centers on MNIST

✓ Auto-encoder Based Data Clustering

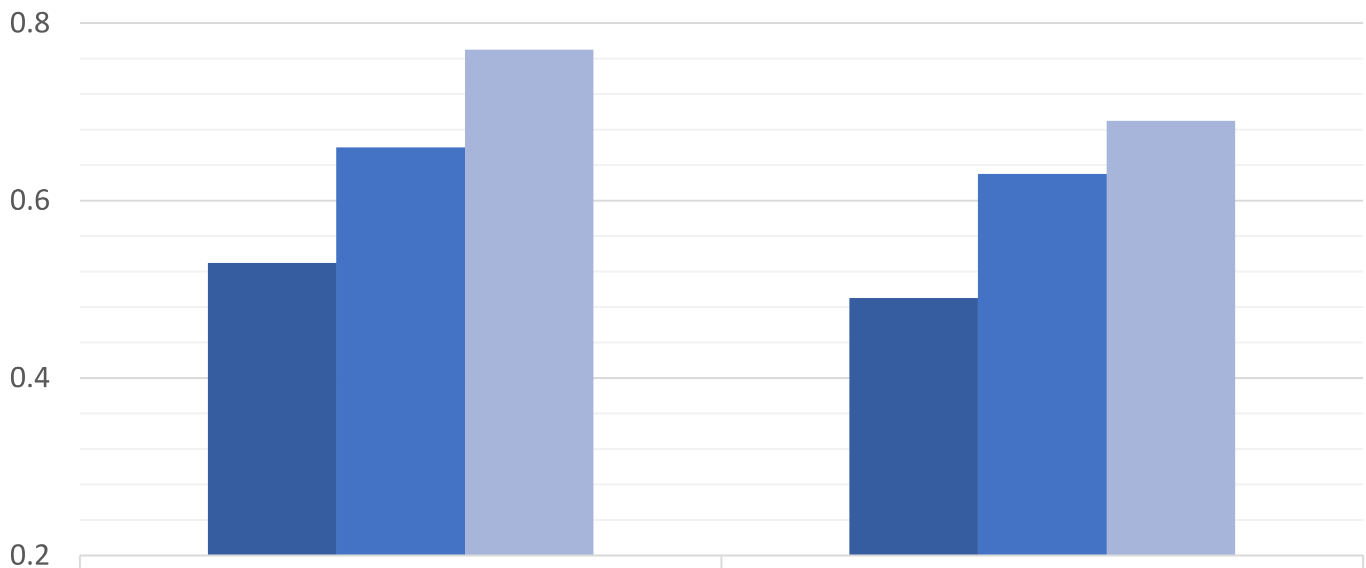


✓ K-means



●Experiments

✓ Performance comparison in three different spaces with k-means



	NMI	ACC
Original	0.53	0.49
Auto-encoder	0.66	0.63
Proposed	0.77	0.69

*Original means the images(pixel) space.
Auto-encoder means the feature space trained by auto-encoder nets.
Proposed means the feature space trained by restrains added auto-encoder nets.

● Conclusions

- ✓ Auto-encoder can provide good non-linear mapping.
- ✓ Auto-encoder nets can provide data-stable network.
- ✓ Restrains added can ensure compact.

● Conclusions

✓ Auto-encoder can provide good non-linear mapping.

✓ Auto-encoder nets can provide data-stable network.

✓ Restrains added can ensure compact.

Is good for clustering.

Thank you!

Any questions?