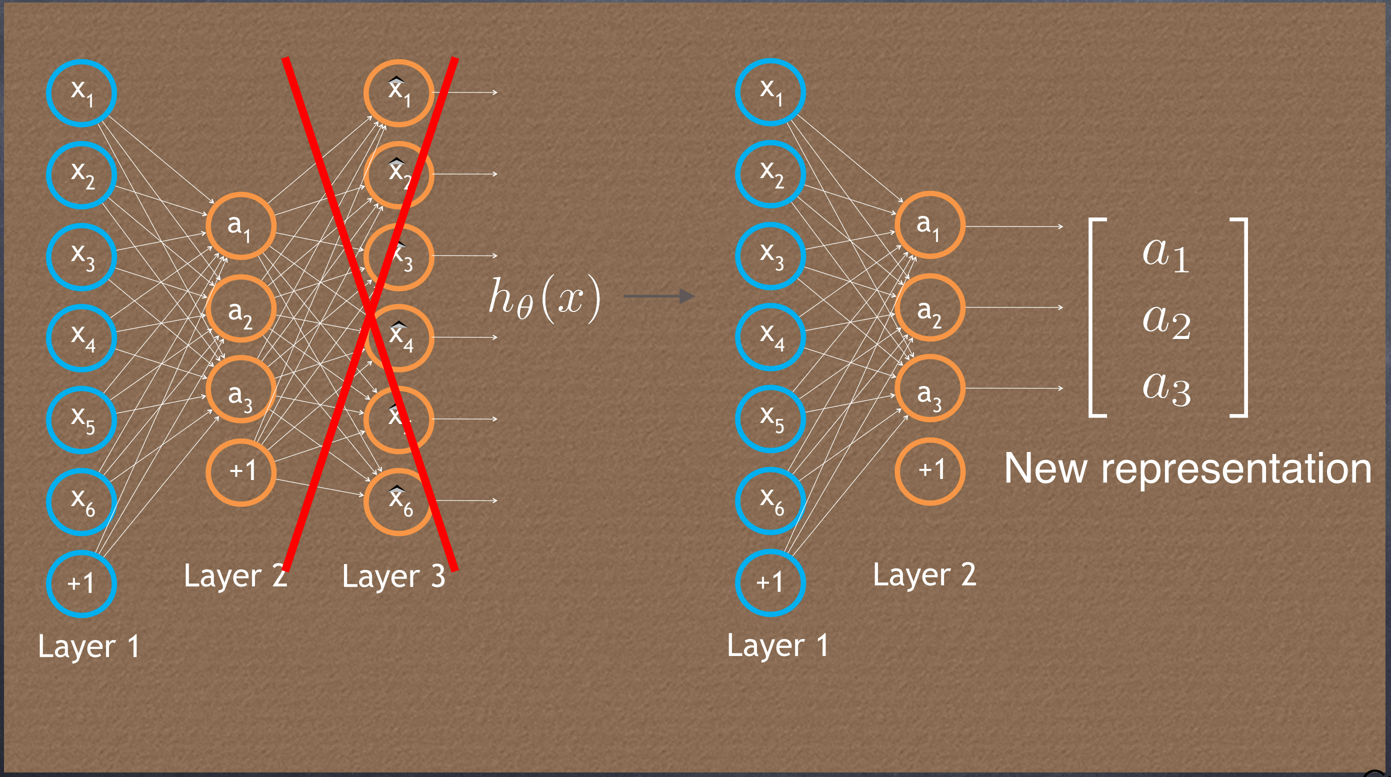
Deep Learning

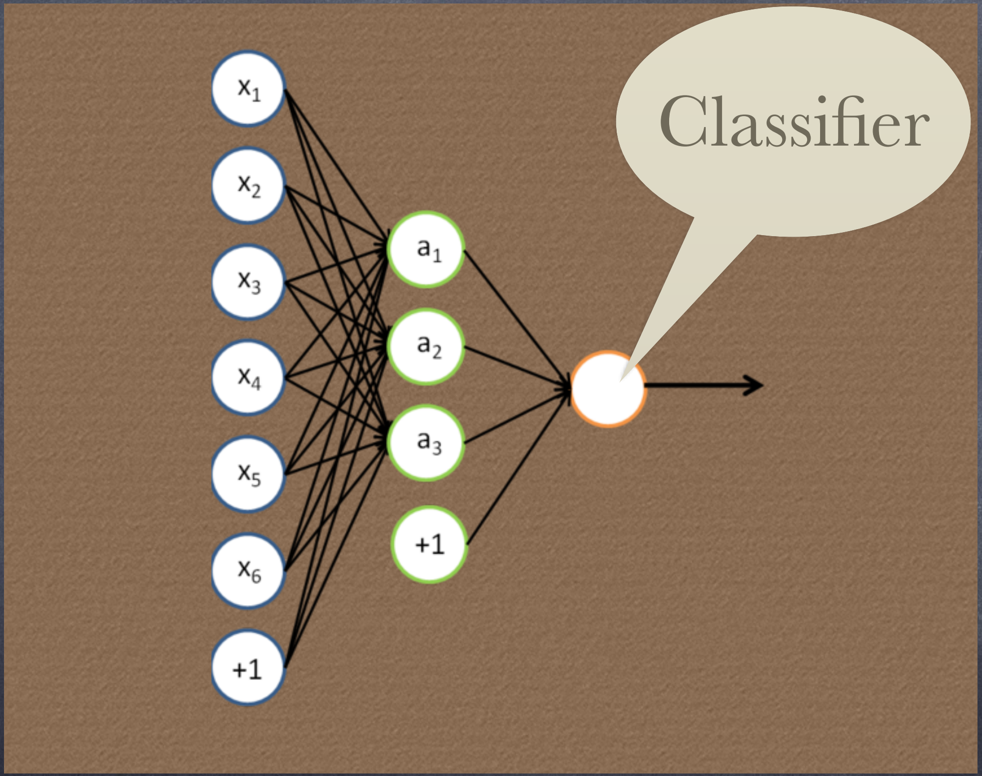
Mayank Vatsa

Autoencoder



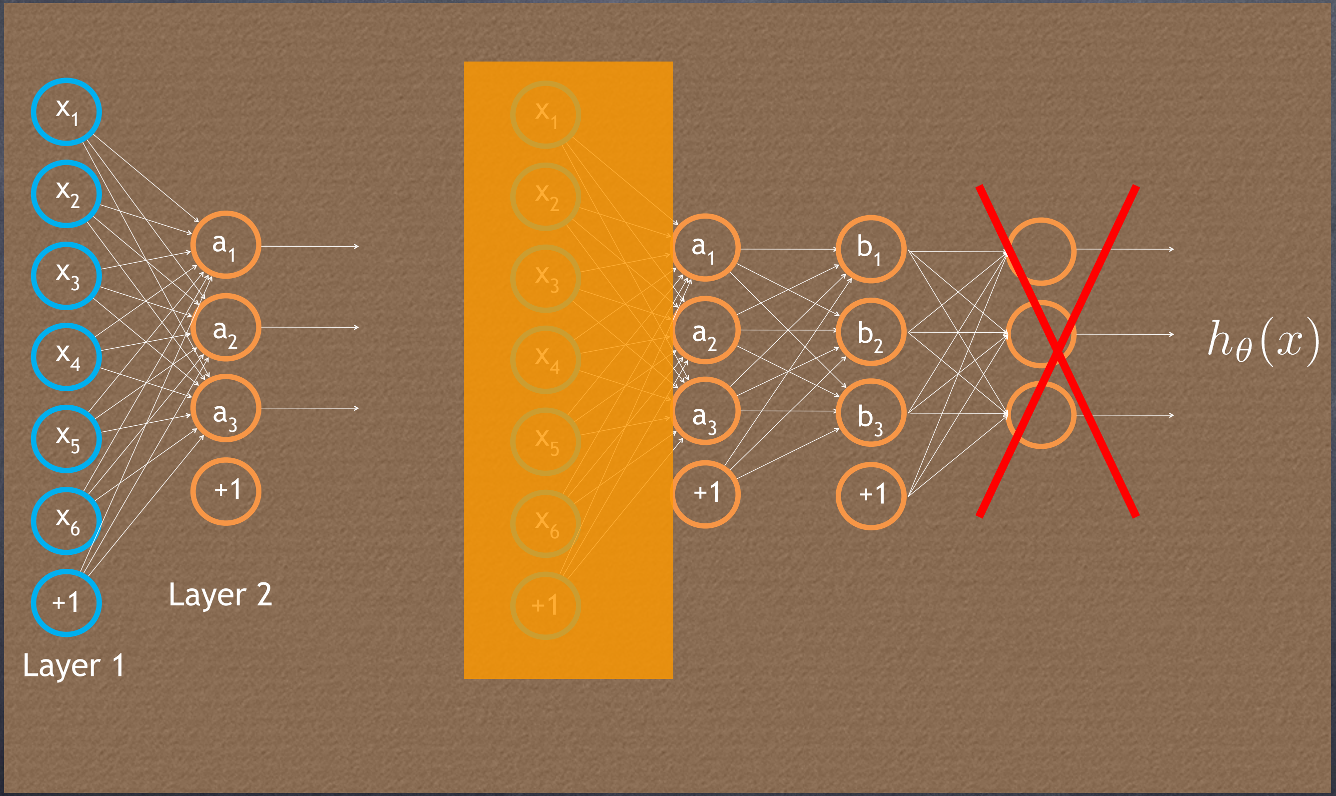
Credit: Andrew Ng

Autoencoder



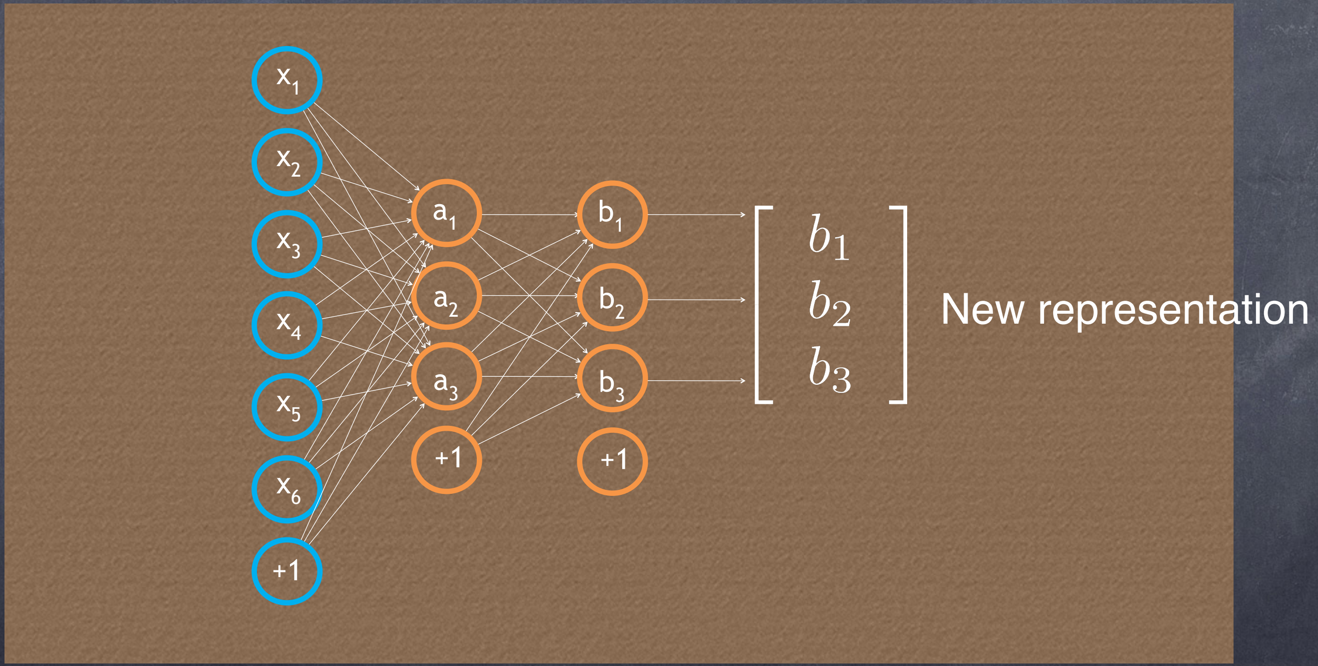
Credit: Andrew Ng

Auto-encoder



Credit: Andrew Ng

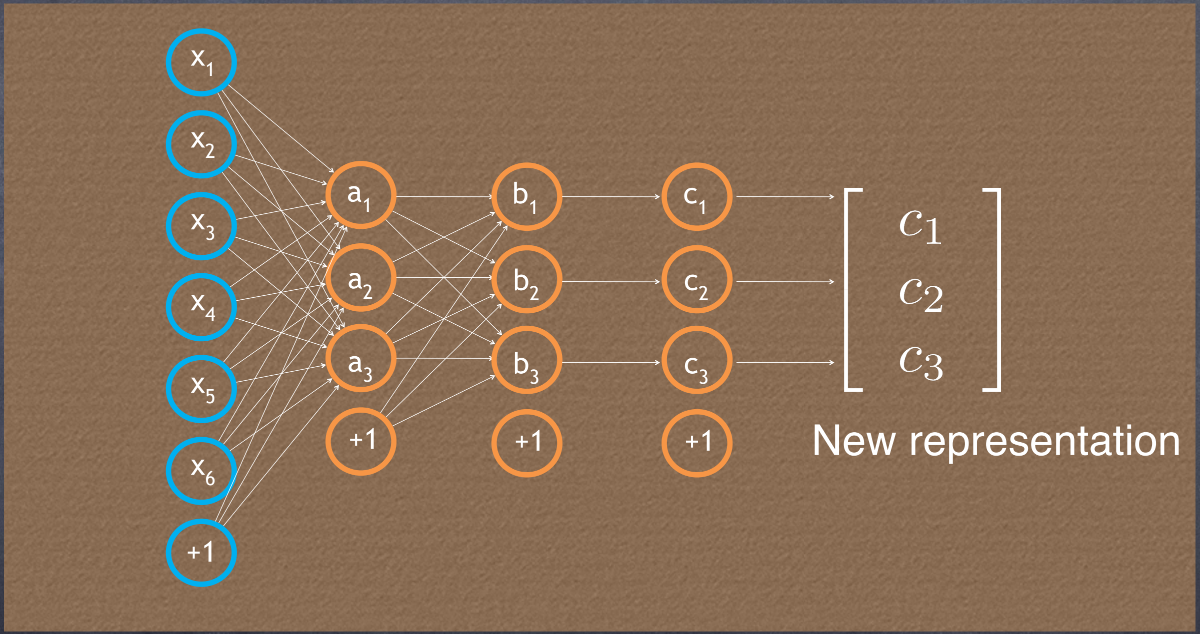
Auto-encoder



Greedy Layer by Layer Training

Credit: Andrew Ng

Multilayer Auto-encoder: Deep Auto-encoder



Credit: Andrew Ng

Multilayer Auto-encoder: Deep Auto-encoder

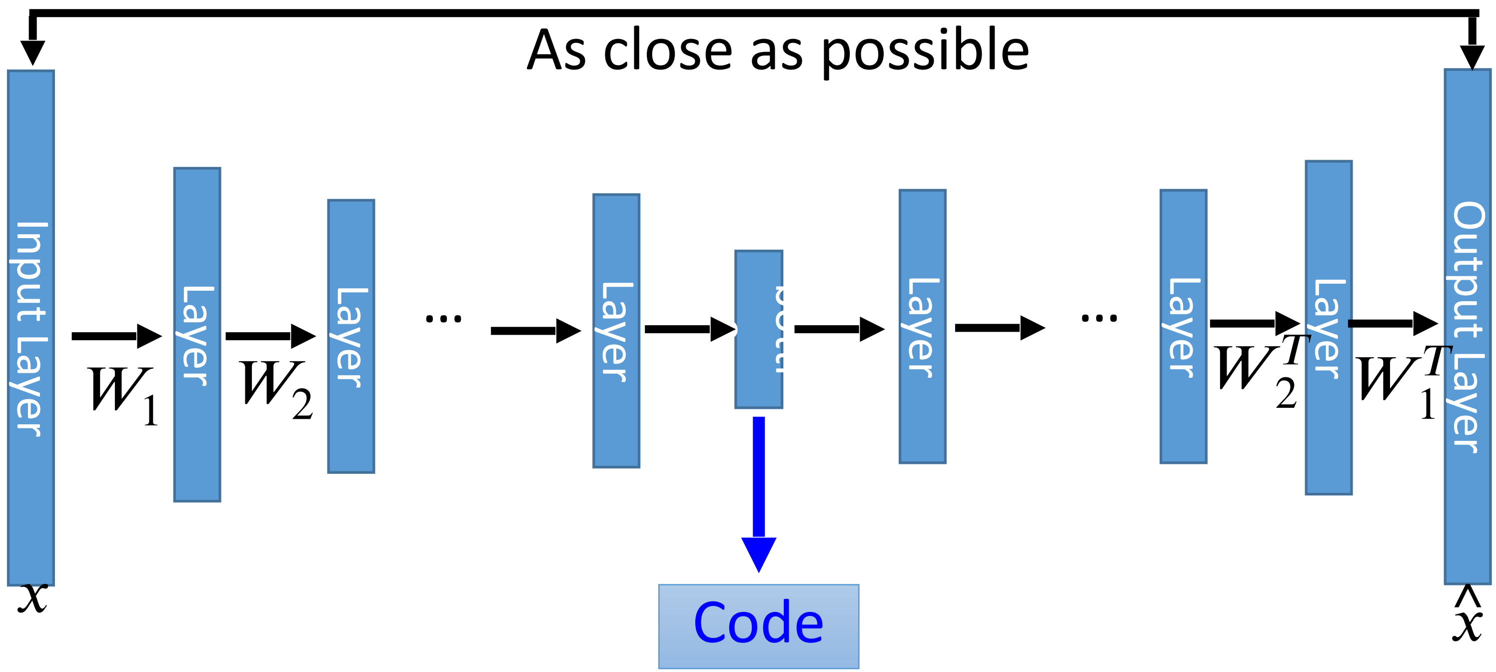
$$\underset{\mathbf{W}}{\text{argmin}}\,||\mathbf{X} - gof(\mathbf{X})||_F^2 + R(\mathbf{W},\mathbf{X}) \;\text{ for all } \mathbf{W}$$

$$g = \mathbf{W}'_1\ \phi(\mathbf{W}'_2 ... \phi(\mathbf{W}'_\mathbf{L}(f(\mathbf{X}))))$$

$$f = \phi(\mathbf{W}_\mathbf{L}\ \phi(\mathbf{W}_{\mathbf{L-1}} ... \phi(\mathbf{W}_1(\mathbf{X}))))$$

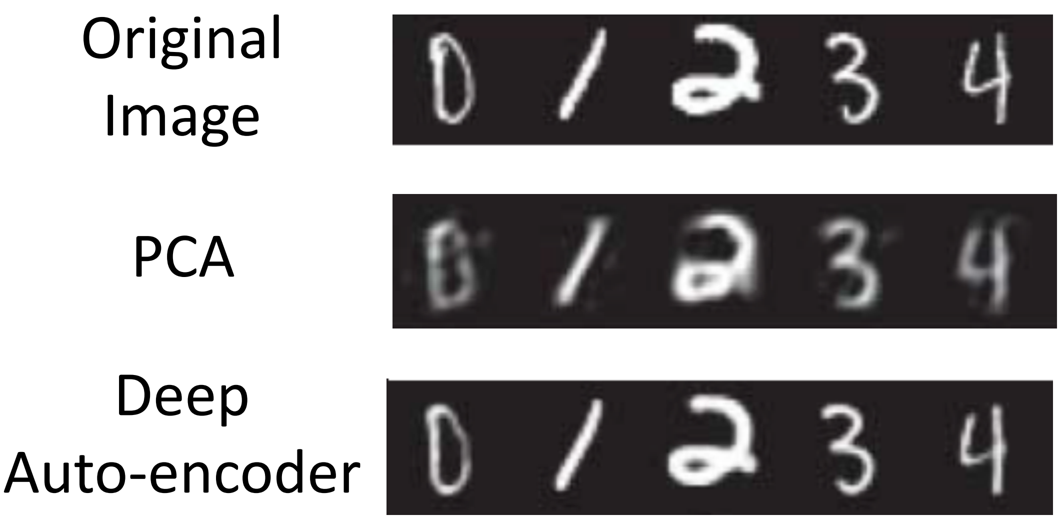
Deep Auto-encoder

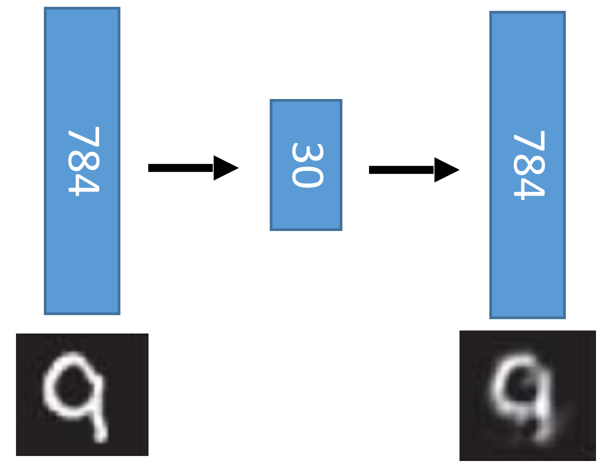
Symmetric is not necessary.

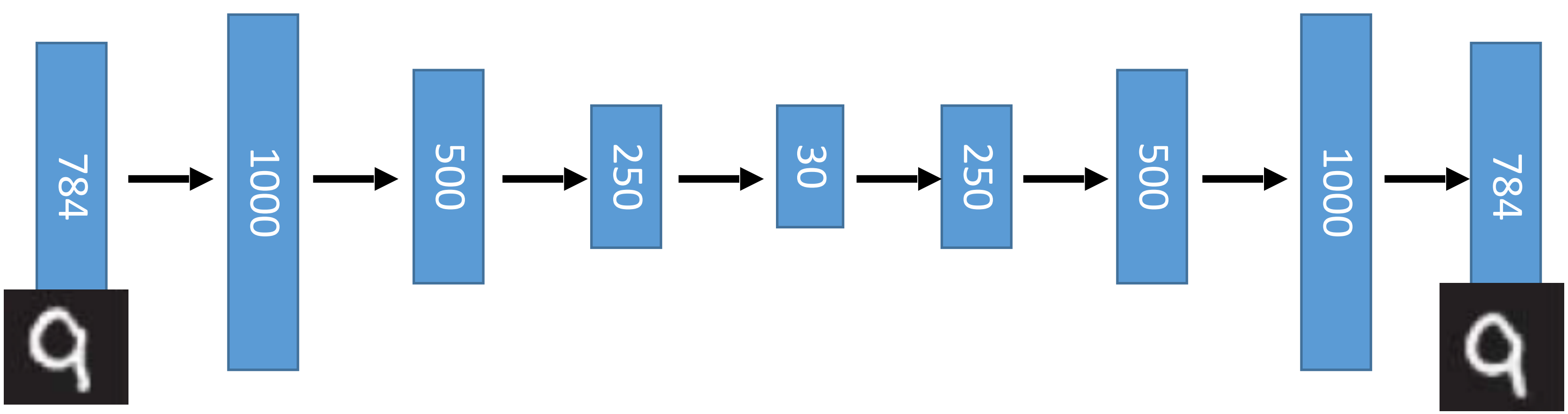


Reference: Hinton, Geoffrey E., and Ruslan R. Salakhutdinov. "Reducing the dimensionality of data with neural networks." *Science* 313.5786 (2006): 504-507

Deep Auto-encoder





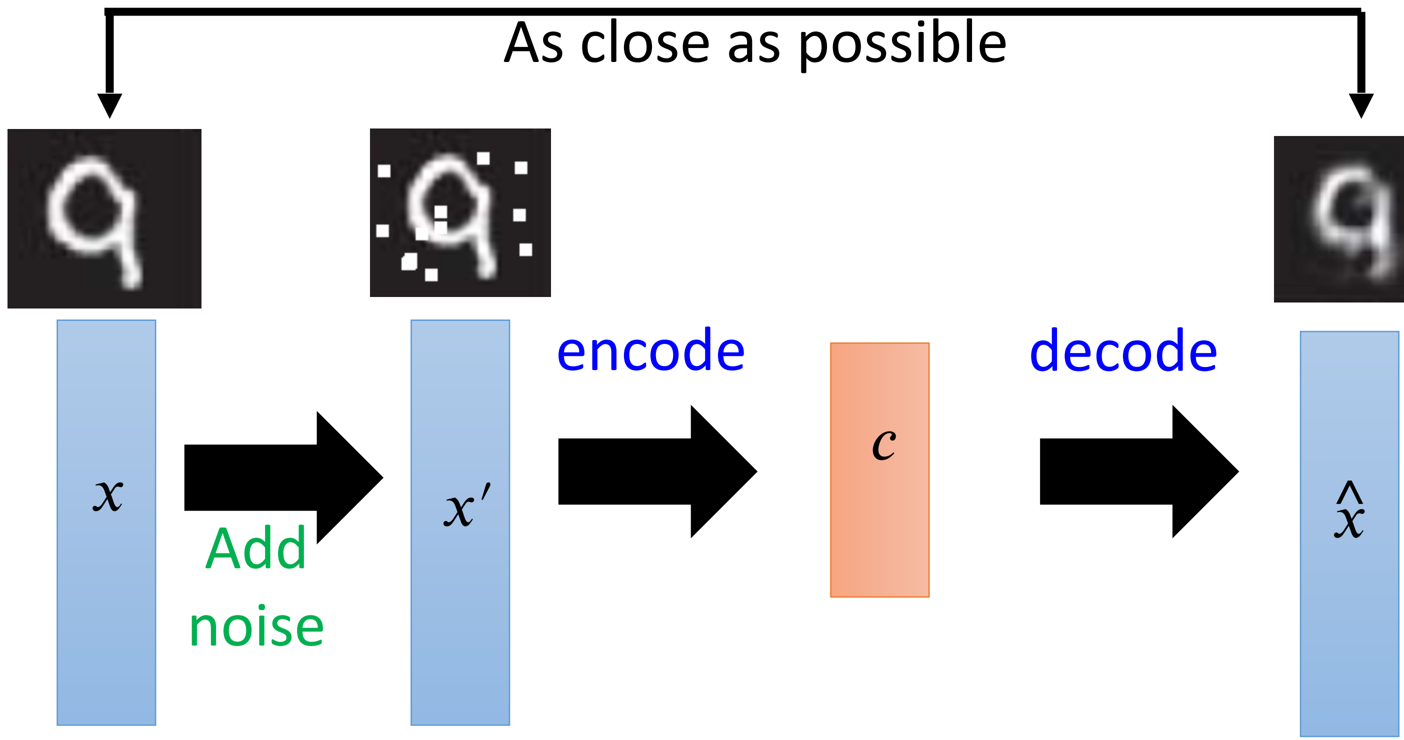


More: Contractive auto-encoder

Auto-encoder

Ref: Rifai, Salah, et al. "Contractive auto-encoders: Explicit invariance during feature extraction.“ *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*. 2011.

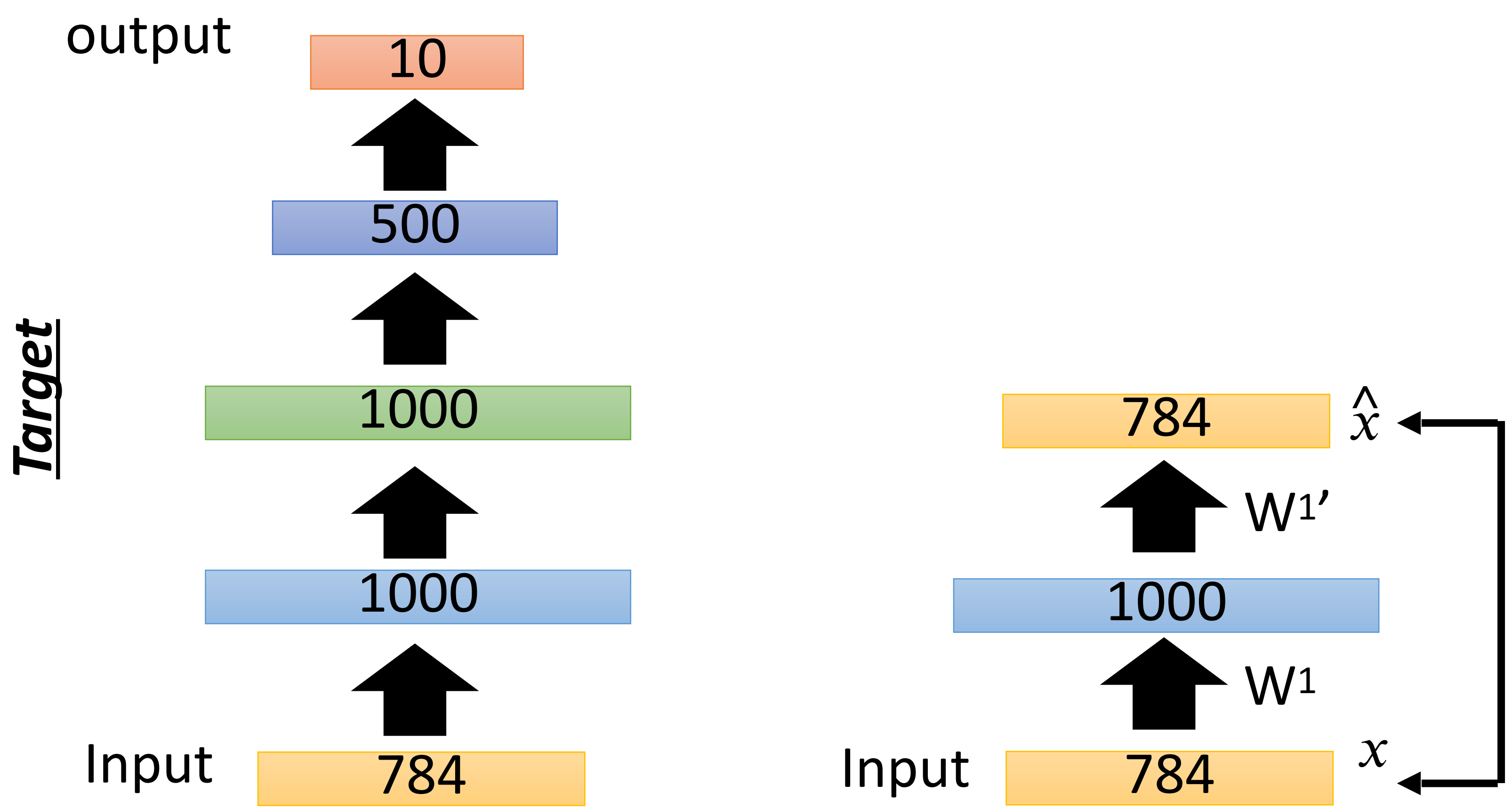
- De-noising auto-encoder



Vincent, Pascal, et al. "Extracting and composing robust features with denoising autoencoders." *ICML,* 2008.

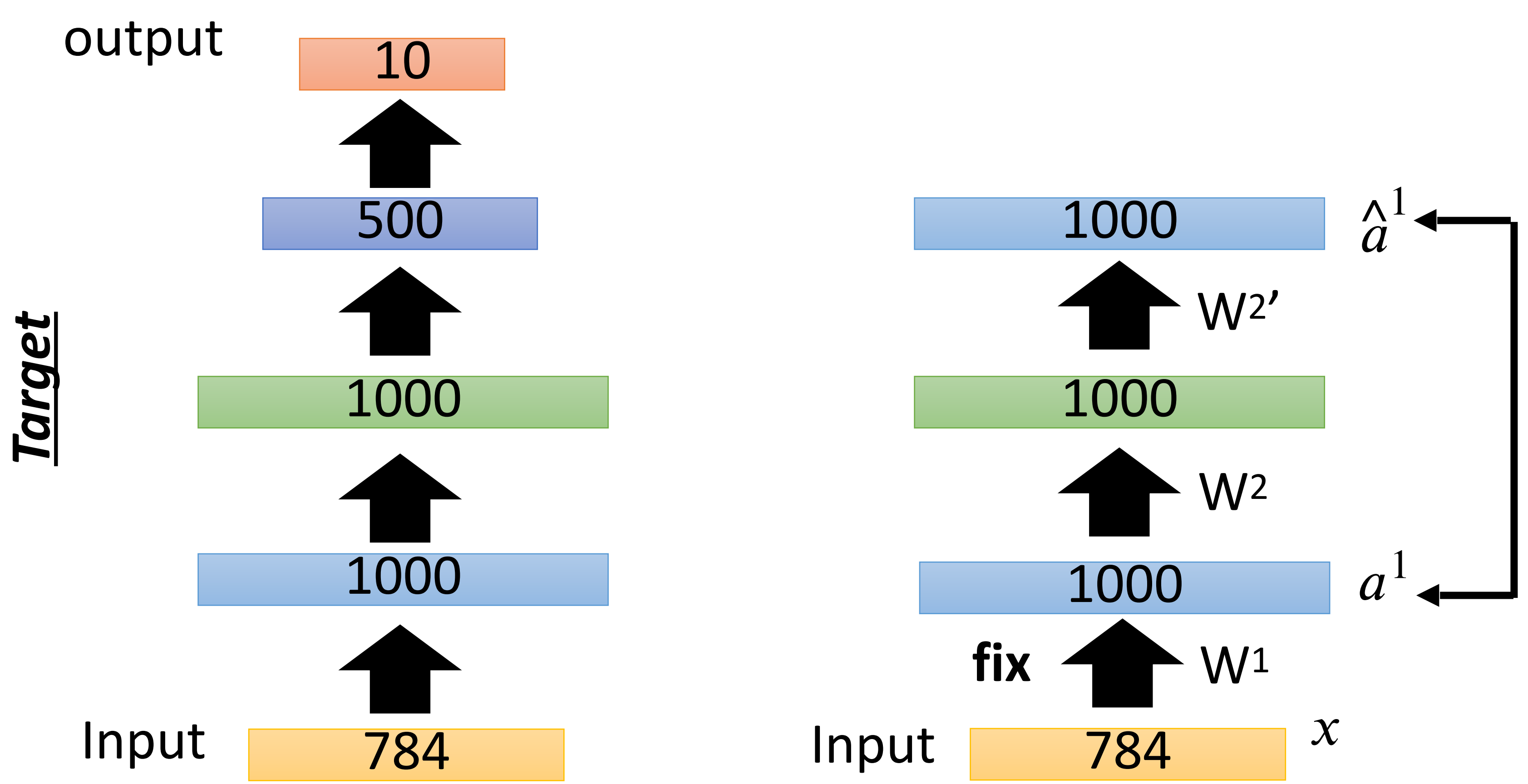
Auto-encoder

- Greedy Layer-wise Pre-training *again*



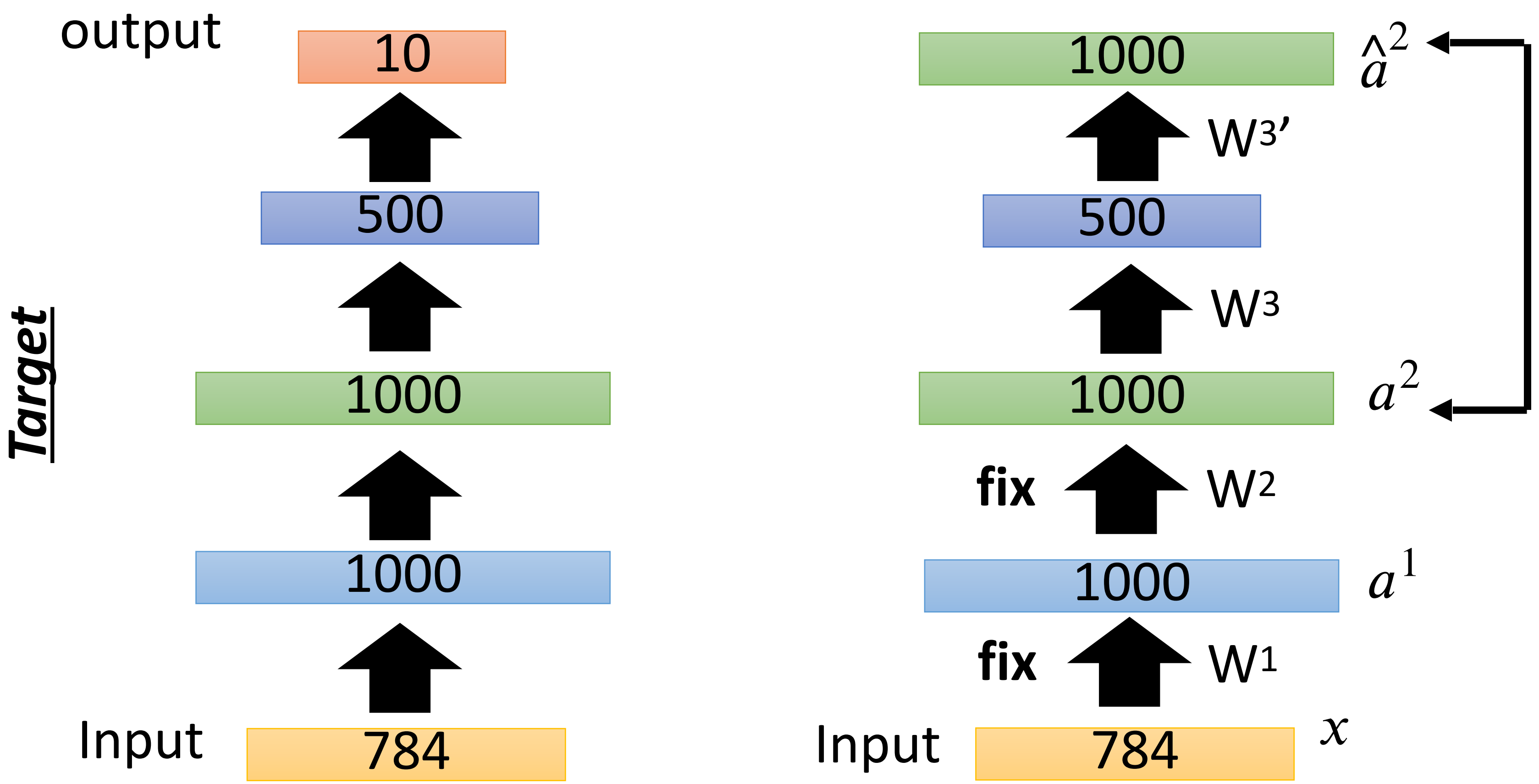
Auto-encoder

- Greedy Layer-wise Pre-training *again*



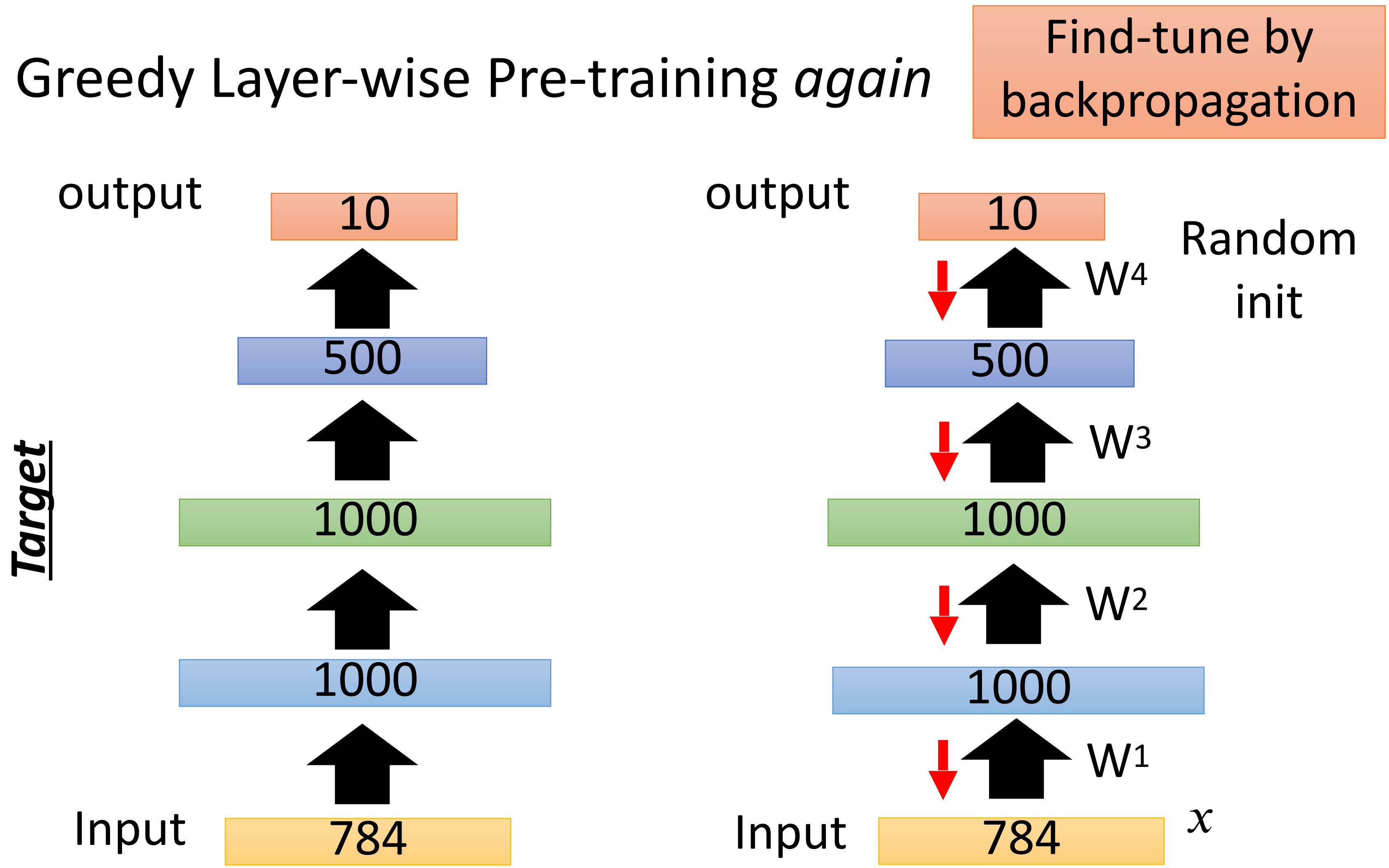
Auto-encoder

- Greedy Layer-wise Pre-training *again*



Auto-encoder

- Greedy Layer-wise Pre-training *again*



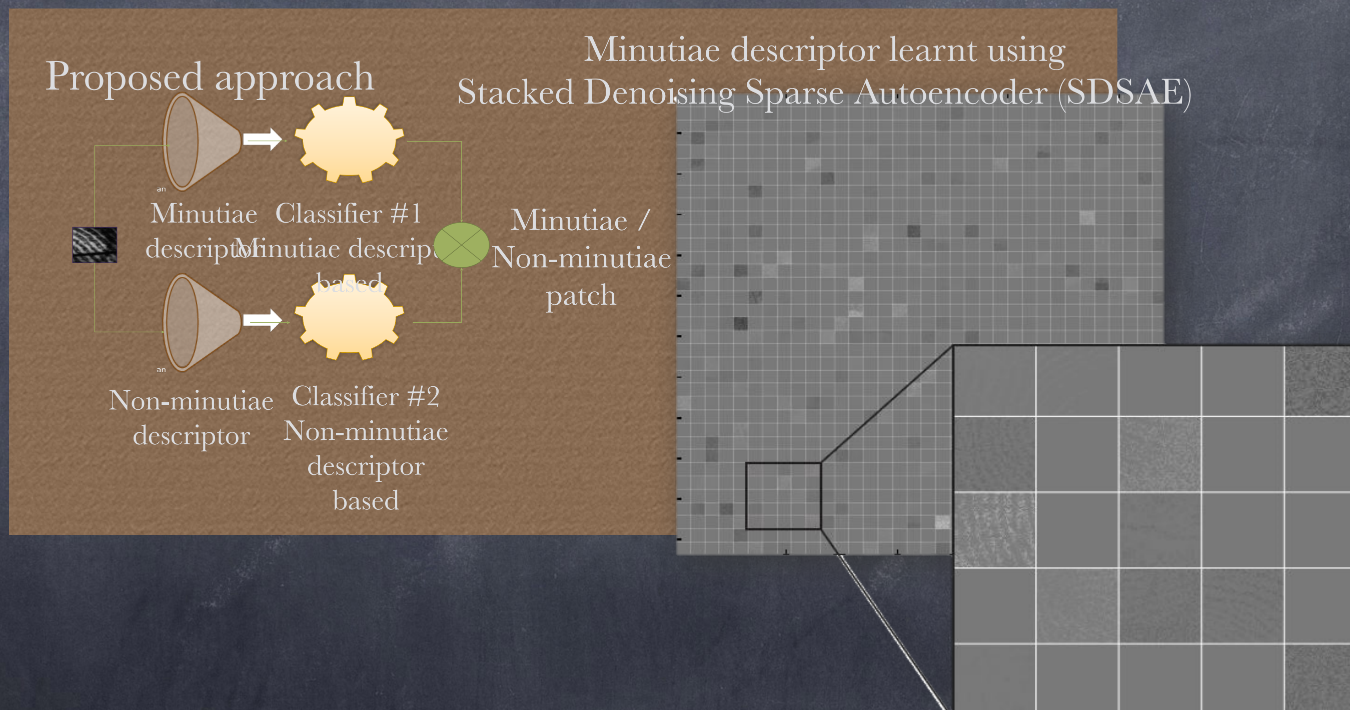
Autoencoders: Applications

- Image colorization: input black and white and train to produce color images

https://www.edureka.co/blog/autoencoders-tutorial/

PURDUE UNIVERSITY

Latent Fingerprint Representation using AE



Sankaran et al. BTAS2

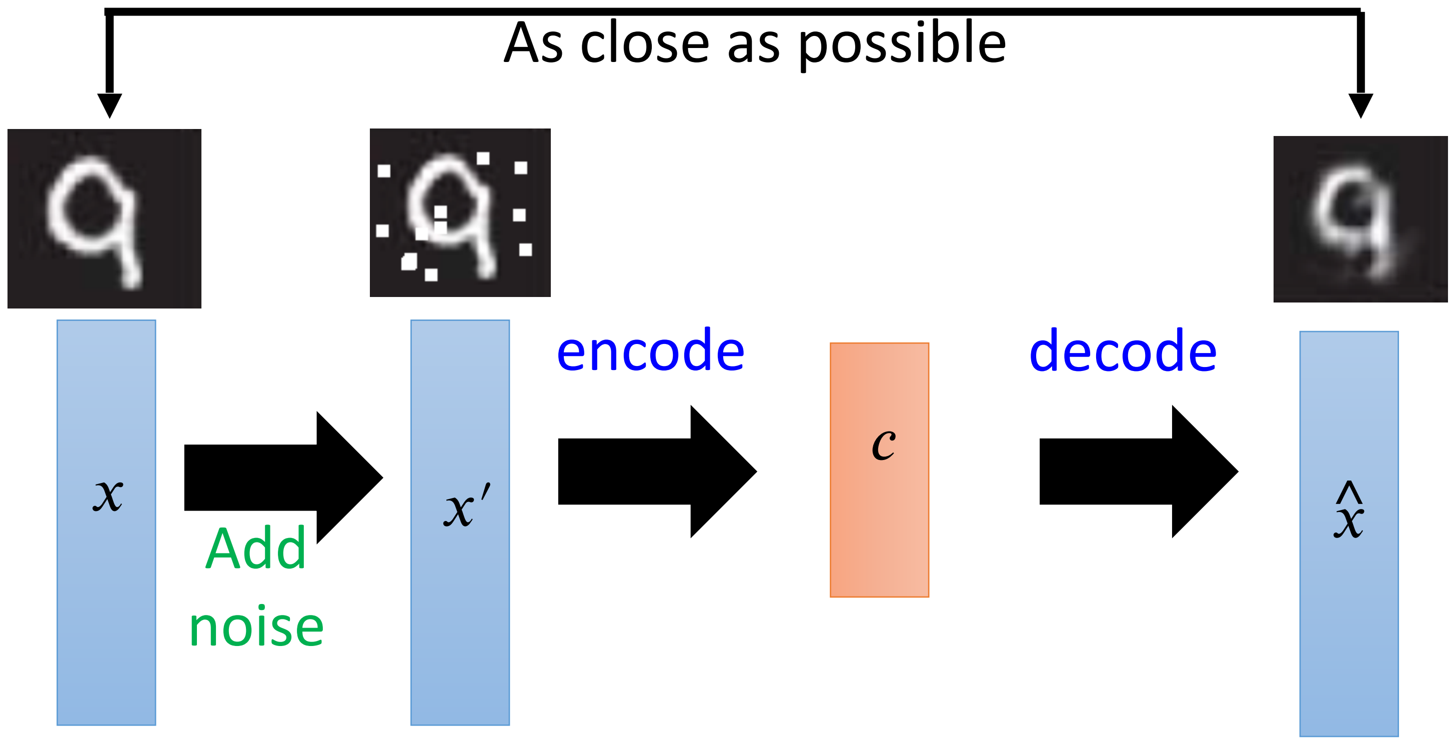
Denoising autoencoders

- Basic autoencoder trains to minimize the loss between x and the reconstruction g(f(x)).
- Denoising autoencoders train to minimize the loss between x and g(f(x+w)), where w is random noise.
- Same possible architectures, different training data.
- [Kaggle has a dataset on damaged documents.]

https://blog.keras.io/building-autoencoders-in-keras.html

Auto-encoder

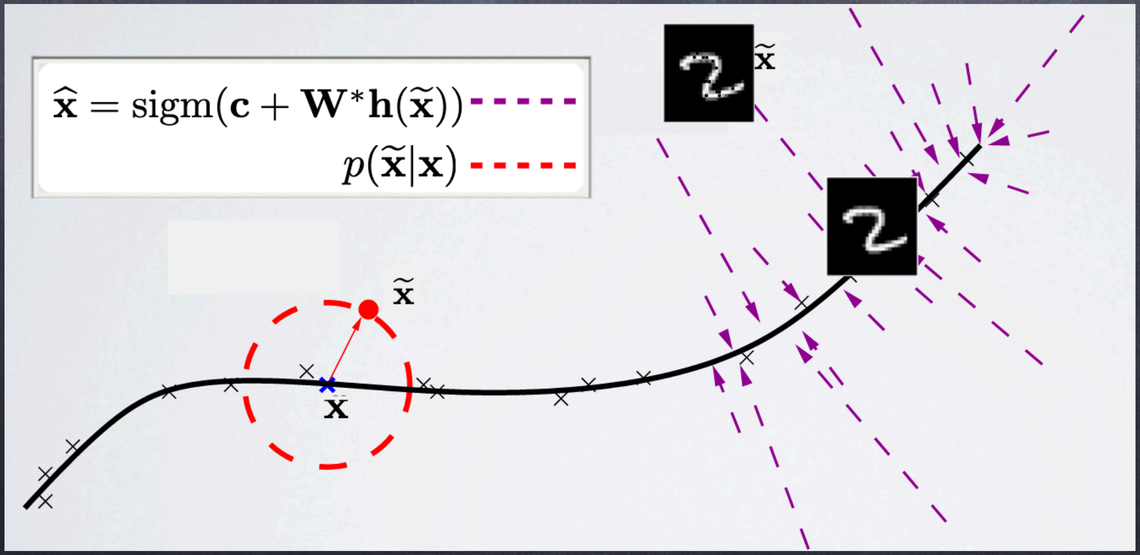
- De-noising auto-encoder



Vincent, Pascal, et al. "Extracting and composing robust features with denoising autoencoders." *ICML,* 2008.

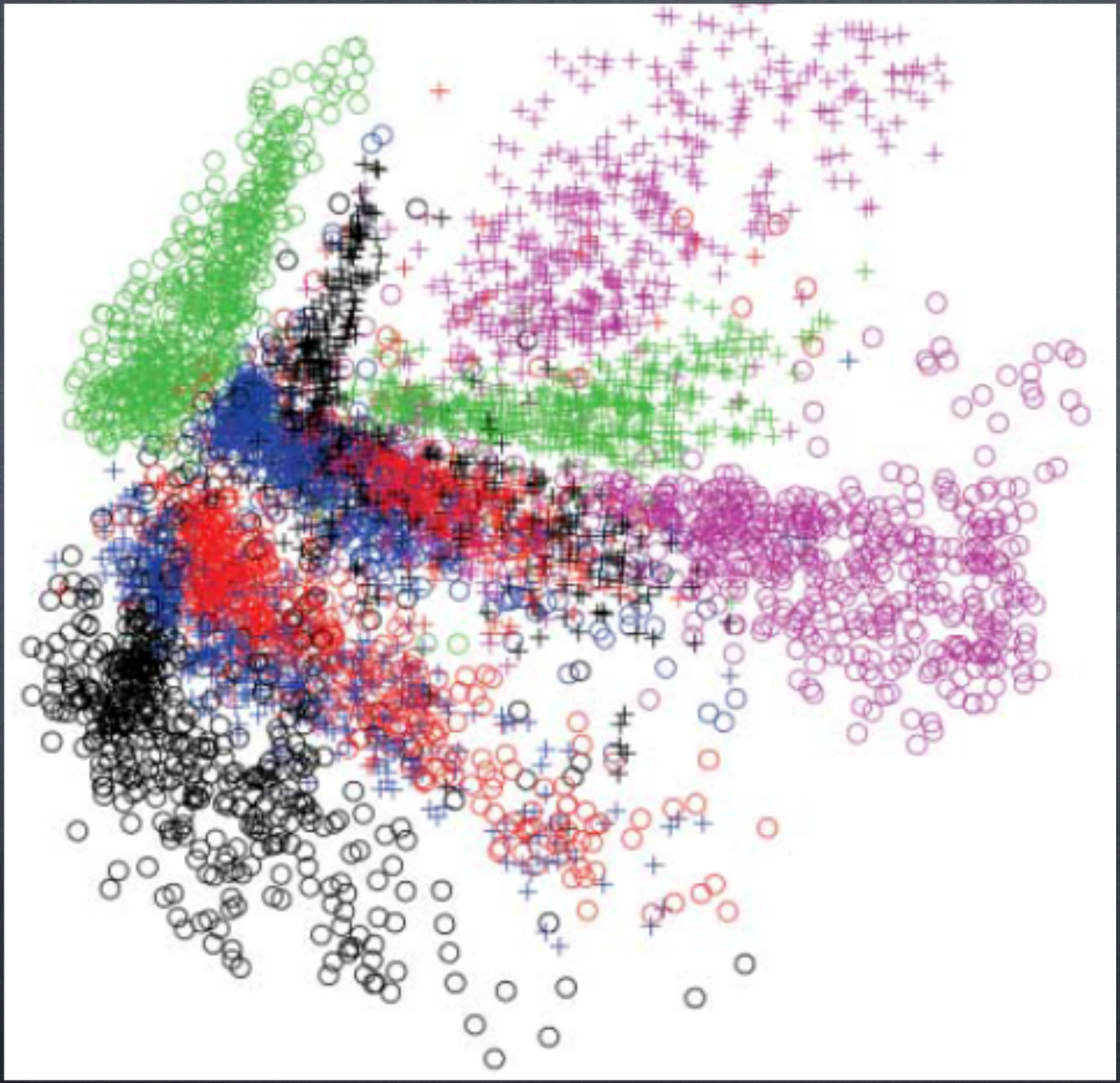
Denoising process is kind of regularization

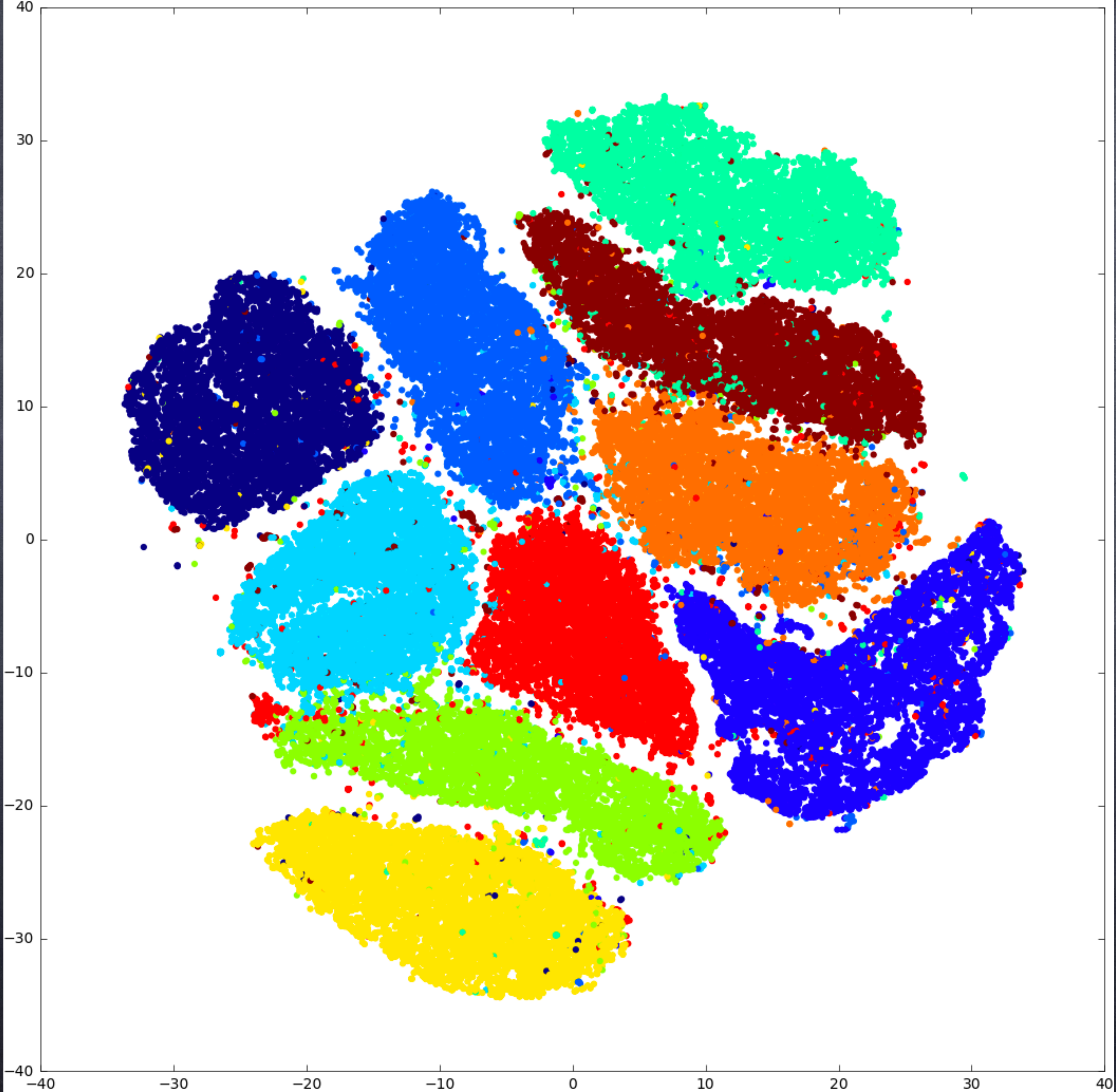
- denoising autoencoder learns a projection from a neighborhood of our training data back onto the training data.



Visualizing Features

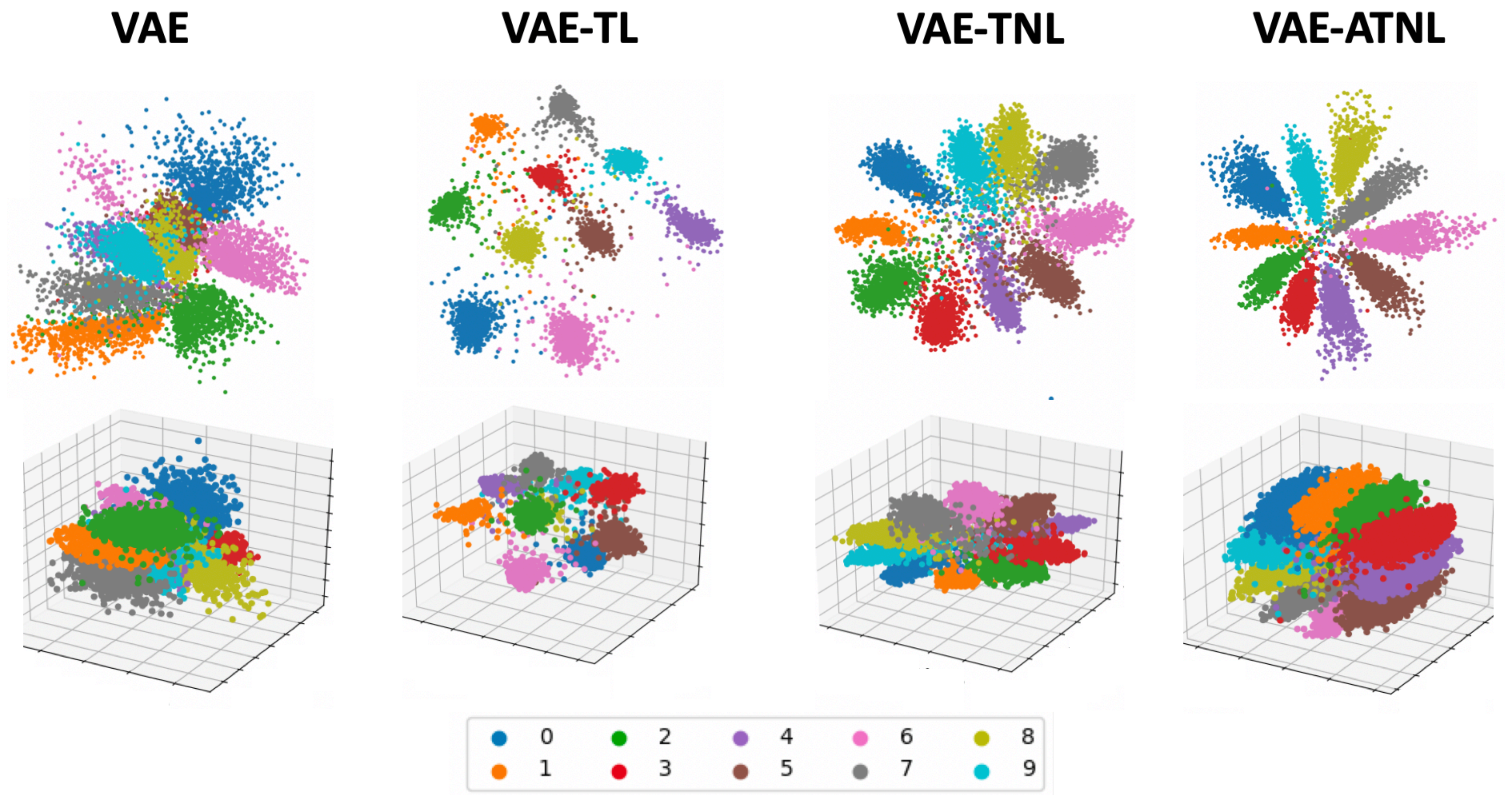
- t-SNE (t-distributed stochastic neighbor embedding)





40
30
20
10
0
−10
−20
−30
−40
−40
−30
−20
−10
0
10
20
30
40

t-SNE as a tool towards explainability

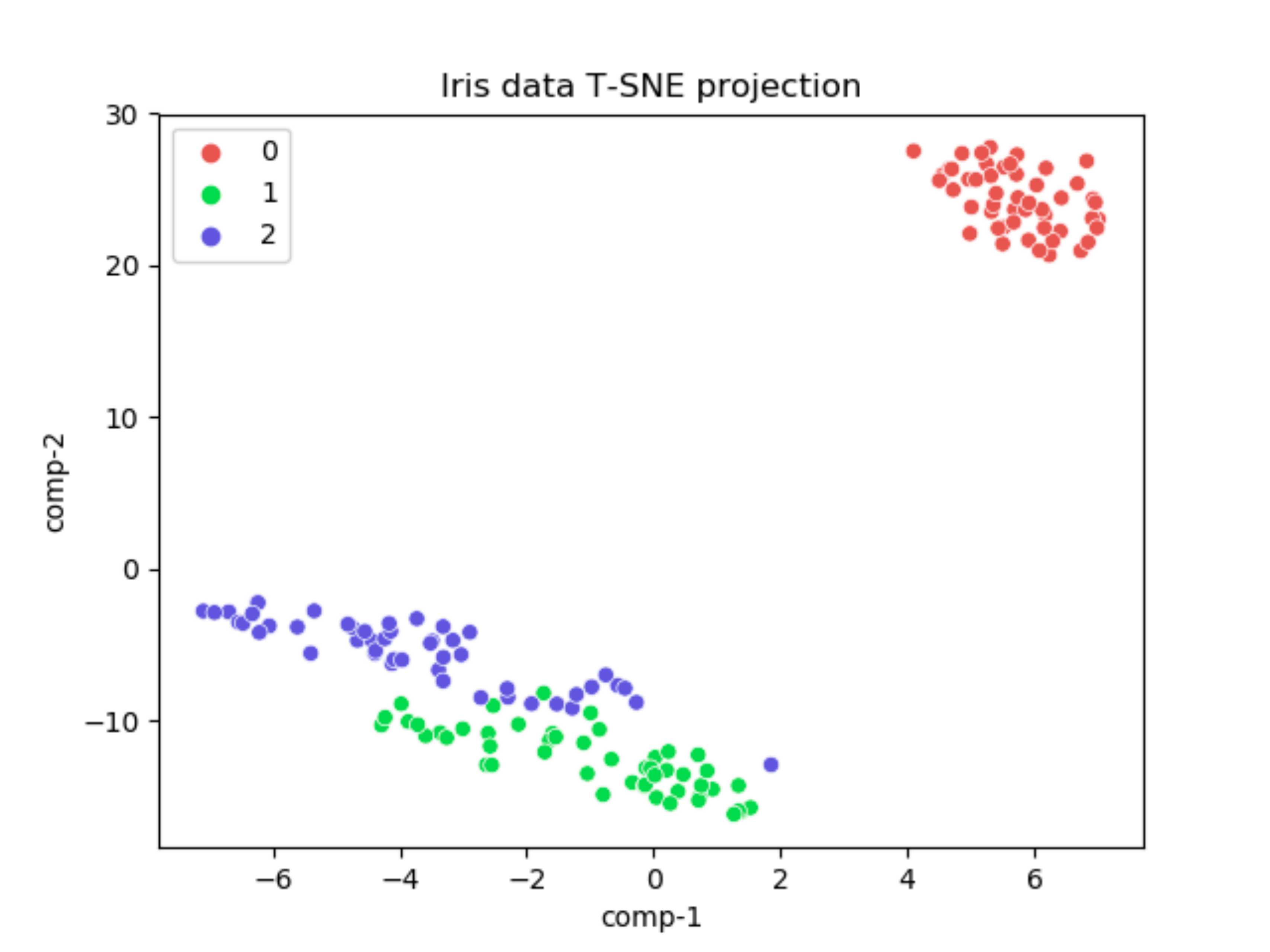


t-SNE as a tool towards explainability

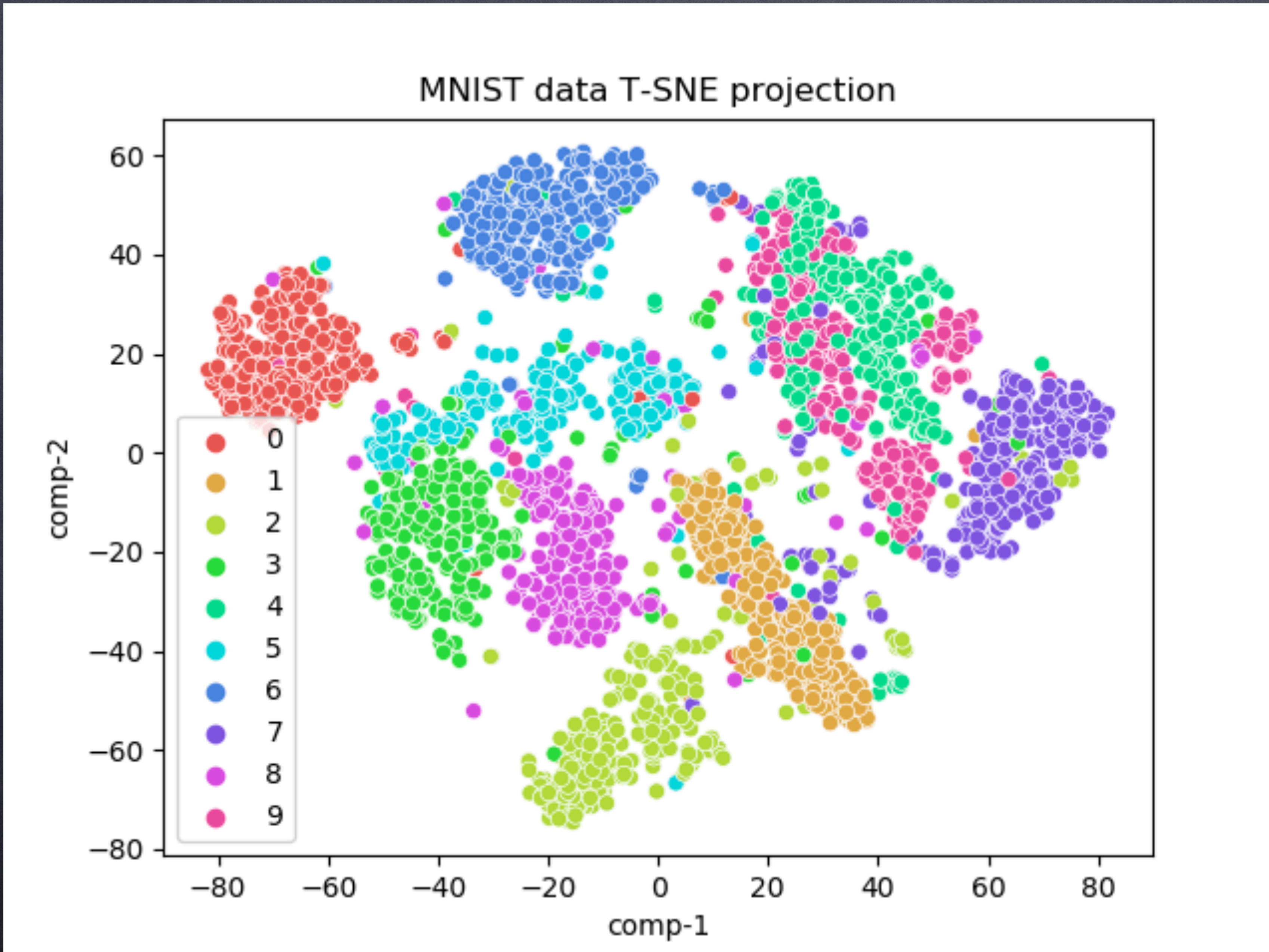
- T-distributed Stochastic Neighbor Embedding (T-SNE) is a tool for visualizing high-dimensional data
- T-SNE is a nonlinear dimensionality reduction technique to visualize data in a two or three dimensional space

Visualizing Iris data

- 1. Load iris data
- 2. Define tSNE using TSNE function - target dim = 2
- z = tsne.fit_transform(.) and then scatter plot

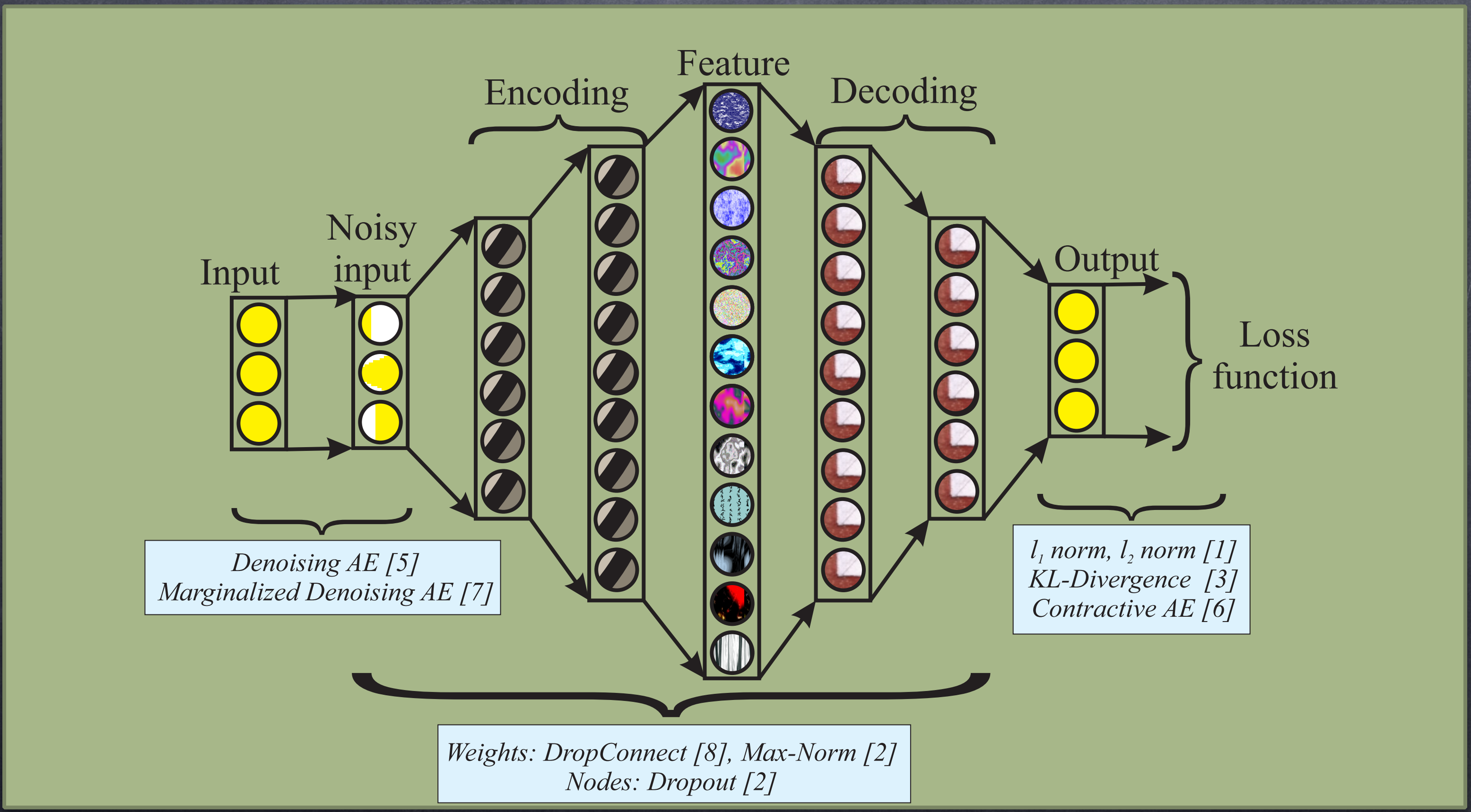


Iris data T-SNE projection
0
1
2
30
20
10
0
−10
−6
−4
−2
0
2
4
6
comp-1
comp-2



MNIST data T-SNE projection
comp-2
comp-1
60
40
20
0
−20
−40
−60
−80
−80
−60
−40
−20
0
20
40
60
80
0
1
2
3
4
5
6
7
8
9

Multilayer Auto-encoder: Regularization Approaches



What would be the generic optimization function?

- Recall AE function