Unsupervised Representation Learning: Autoencoders and Factorization

Ju Sun
Computer Science & Engineering
University of Minnesota, Twin Cities

October 25, 2022

Recap & preview

We have talked about

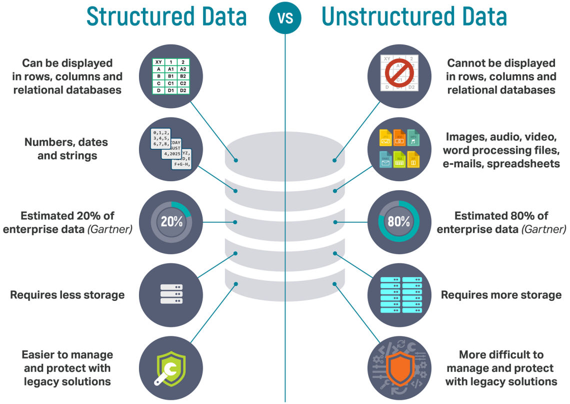
- Basic DNNs (multi-layer feedforward)
- Universal approximation theorems
- Basics of numerical optimization
- Training DNNs: basic methods and tricks

Models and Applications

- Unsupervised representation learning: autoencoders and variants
- DNNs for grid data: CNNs
- DNNs for sequential data: RNNs
- DNNs for graph data: GNNs
- Generative models: GAN, VAE, normalization flow, diffusion models
- Interactive models: reinforcement learning
- Self-supervised learning (if time permits)

involve modification and composition of the basic DNNs

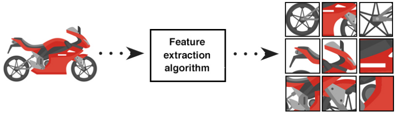
Structured vs. unstructured data



Credit: https://lawtomated.com/structured-data-vs-unstructured-data-what-are-they-and-why-care/

- structured data also called tabular data
- structured data often directly fed into classical ML tools
- the success of DL mostly lies at learning useful features/patterns from unstructured data, i.e., **representation learning**

Feature engineering for unstructured data: old and new



Credit: [Elgendy, 2020]

Feature engineering: derive features for efficient learning

Traditional learning pipeline

- feature extraction is “independent” of the learning models and tasks
- features are handcrafted and/or learned

Modern learning pipeline

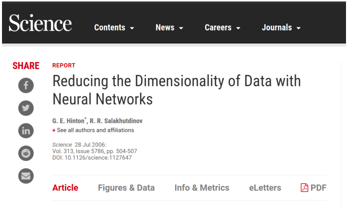
- end-to-end DNN learning

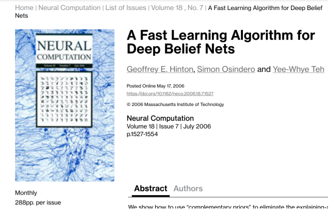
Unsupervised representation learning

Learning feature/representation without task information (e.g., labels)
(ICLR — International Conference on Learning Representation)

Why not jump into the end-to-end learning?

- **Historical**: Unsupervised representation learning key to the revival of deep learning (i.e., layerwise pretraining, [Hinton et al., 2006, Hinton, 2006])





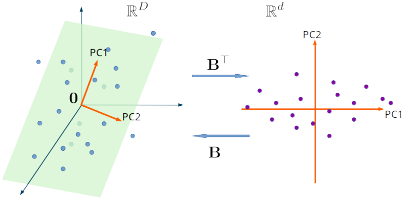
- **Practical**: Numerous advanced models built on top of the ideas in unsupervised representation learning (e.g., encoder-decoder networks)

Outline

PCA: the geometric picture

Principal component analysis (PCA)

- $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \in \mathbb{R}^D$ zero-centered and write $\boldsymbol{X} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m]^\intercal \in \mathbb{R}^{m \times D}$
- Compact SVD $\boldsymbol{X} = \boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^\intercal$, where $\boldsymbol{V} \in \mathbb{R}^{D \times r}$ spans the row space of $\boldsymbol{X}$
- Take top right singular vectors $\boldsymbol{B}$ from $\boldsymbol{V}$, and obtain $\boldsymbol{X}\boldsymbol{B}$

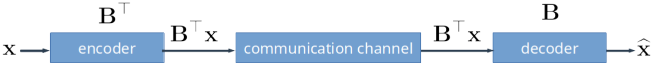


PCA is effectively to identify the best-fit subspace to $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m$

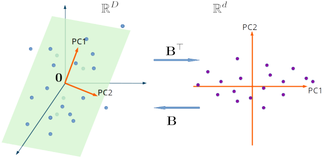
- $\boldsymbol{B}$ has orthonormal columns, i.e., $\boldsymbol{B}^\intercal\boldsymbol{B} = \boldsymbol{I}$ ($\boldsymbol{B}\boldsymbol{B}^\intercal \neq \boldsymbol{I}$ when $D \neq d$)
- sample to representation: $\boldsymbol{x} \mapsto \boldsymbol{x}' \doteq \boldsymbol{B}^\intercal\boldsymbol{x}$ ($\mathbb{R}^D \to \mathbb{R}^d$, dimension reduction)
- representation to sample: $\boldsymbol{x}' \mapsto \widehat{\boldsymbol{x}} \doteq \boldsymbol{B}\boldsymbol{x}'$ ($\mathbb{R}^d \to \mathbb{R}^D$)
- $\widehat{\boldsymbol{x}} = \boldsymbol{B}\boldsymbol{B}^\intercal\boldsymbol{x} \approx \boldsymbol{x}$

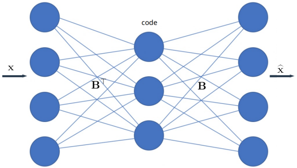
Autoencoders

story in digital communications ...



autoencoder: [Bourlard and Kamp, 1988, Hinton and Zemel, 1994]





– **Encoding**:
$\boldsymbol{x} \mapsto \boldsymbol{x}' = \boldsymbol{B}^\intercal \boldsymbol{x}$

– **Decoding**:
$\boldsymbol{x}' \mapsto \boldsymbol{B}\boldsymbol{B}^\intercal \boldsymbol{x} = \widehat{\boldsymbol{x}}$

To find the basis $\boldsymbol{B}$, solve $(d \leq D)$

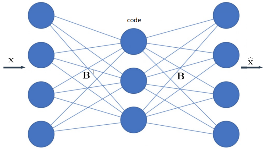
$$\min_{\boldsymbol{B} \in \mathbb{R}^{D \times d}} \sum_{i=1}^{m} \|\boldsymbol{x}_i - \boldsymbol{B}\boldsymbol{B}^\intercal \boldsymbol{x}_i\|_2^2$$

or:

$$\min_{\boldsymbol{B} \in \mathbb{R}^{D \times d}} \|\boldsymbol{X}^\intercal - \boldsymbol{B}\boldsymbol{B}^\intercal \boldsymbol{X}^\intercal\|_F^2$$

Autoencoders

autoencoder:



To find the basis $\boldsymbol{B}$, solve

$$\min_{\boldsymbol{B}\in\mathbb{R}^{D\times d}} \sum_{i=1}^{m} \|\boldsymbol{x}_i - \boldsymbol{B}\boldsymbol{B}^\intercal \boldsymbol{x}_i\|_2^2$$

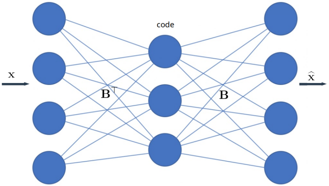
So the autoencoder is performing PCA!

One can even relax the weight tying:

$$\min_{\boldsymbol{B}\in\mathbb{R}^{D\times d}, \boldsymbol{A}\in\mathbb{R}^{D\times d}} \sum_{i=1}^{m} \|\boldsymbol{x}_i - \boldsymbol{B}\boldsymbol{A}^\intercal \boldsymbol{x}_i\|_2^2 \,,$$

which finds a basis (not necessarily orthonormal) $\boldsymbol{B}$ that spans the top singular space also [Baldi and Hornik, 1989], [Kawaguchi, 2016], [Lu and Kawaguchi, 2017].

Factorization



To perform PCA,

$$\min_{\boldsymbol{B}\in\mathbb{R}^{D\times d}} \sum_{i=1}^{m} \|\boldsymbol{x}_i - \boldsymbol{B}\boldsymbol{B}^\intercal \boldsymbol{x}_i\|_2^2$$

$$\min_{\boldsymbol{B}\in\mathbb{R}^{D\times d}, \boldsymbol{A}\in\mathbb{R}^{D\times d}} \sum_{i=1}^{m} \|\boldsymbol{x}_i - \boldsymbol{B}\boldsymbol{A}^\intercal \boldsymbol{x}_i\|_2^2 ,$$

But: the basis $\boldsymbol{B}$ and the representations/codes $\boldsymbol{z}_i$'s are all we care about

Factorization: (or autoencoder without encoder)

$$\min_{\boldsymbol{B}\in\mathbb{R}^{D\times d}, \boldsymbol{z}_i' s\in\mathbb{R}^d} \sum_{i=1}^{m} \|\boldsymbol{x}_i - \boldsymbol{B}\boldsymbol{z}_i\|_2^2 .$$

All three formulations will find three different $\boldsymbol{B}$'s that span the same principal subspace [Tan and Mayrovouniotis, 1995, Li et al., 2020b, Li et al., 2020a, Valavi et al., 2020]. They're all doing PCA!

Sparse coding

Factorization: (or autoencoder without encoder)

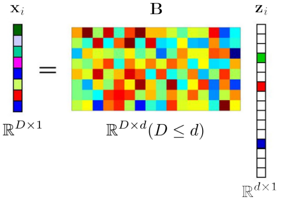
$$\min_{\boldsymbol{B}\in\mathbb{R}^{D\times d},\boldsymbol{z}_i's\in\mathbb{R}^d} \sum_{i=1}^{m} \|\boldsymbol{x}_i - \boldsymbol{B}\boldsymbol{z}_i\|_2^2 .$$

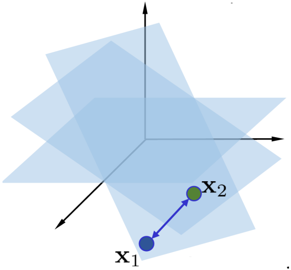
What happens when we allow $d \geq D$? Underdetermined even if $\boldsymbol{B}$ is known.

Sparse coding: assuming $\boldsymbol{z}_i$'s are sparse and $d \geq D$

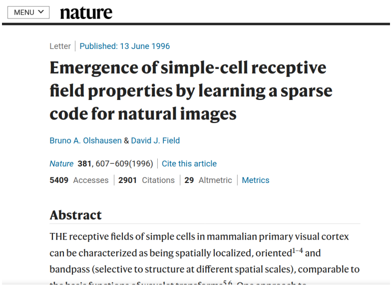
$$\min_{\boldsymbol{B}\in\mathbb{R}^{D\times d},\boldsymbol{z}_i's\in\mathbb{R}^d} \sum_{i=1}^{m} \|\boldsymbol{x}_i - \boldsymbol{B}\boldsymbol{z}_i\|_2^2 + \lambda \sum_{i=1}^{m} \Omega\left(\boldsymbol{z}_i\right)$$

where Ω promotes sparsity, e.g., $\Omega = \|\cdot\|_1$.





More on sparse coding


MENU nature

Letter | Published: 13 June 1996

Emergence of simple-cell receptive field properties by learning a sparse code for natural images

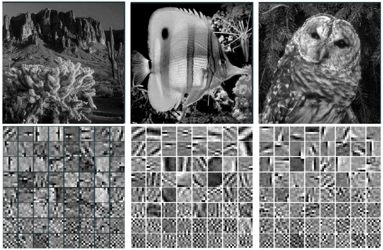
Bruno A. Olshausen & David J. Field

Nature **381**, 607–609(1996) | Cite this article

5409 Accesses | **2901** Citations | **29** Altmetric | Metrics

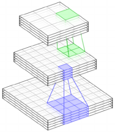
Abstract

THE receptive fields of simple cells in mammalian primary visual cortex can be characterized as being spatially localized, oriented[1–4] and bandpass (selective to structure at different spatial scales), comparable to



denoising

super resol.

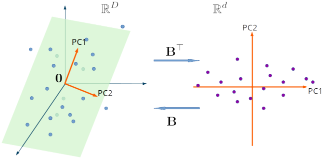


recognition

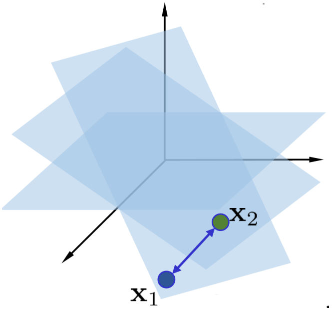
also known as **(sparse) dictionary learning** [Olshausen and Field, 1996, Mairal, 2014, Sun et al., 2017, Bai et al., 2018, Qu et al., 2019]

Outline

Quick summary of the linear models



PCA is effectively to identify the best-fit subspace to $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m$

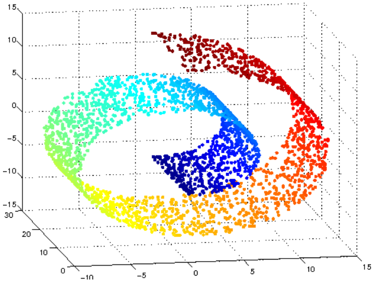


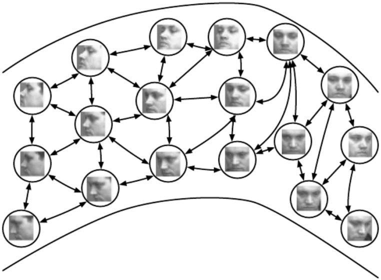
- $\boldsymbol{B}$ from $\boldsymbol{V}$ of $\boldsymbol{X} = \boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^\intercal$
- autoencoder: $\min_{\boldsymbol{B} \in \mathbb{R}^{D \times d}} \sum_{i=1}^m \|\boldsymbol{x}_i - \boldsymbol{B}\boldsymbol{B}^\intercal \boldsymbol{x}_i\|_2^2$
- autoencoder: $\min_{\boldsymbol{B} \in \mathbb{R}^{D \times d}, \boldsymbol{A} \in \mathbb{R}^{D \times d}} \sum_{i=1}^m \|\boldsymbol{x}_i - \boldsymbol{B}\boldsymbol{A}^\intercal \boldsymbol{x}_i\|_2^2$
- factorization: $\min_{\boldsymbol{B} \in \mathbb{R}^{D \times d}, \boldsymbol{z}_i' s \in \mathbb{R}^d} \sum_{i=1}^m \|\boldsymbol{x}_i - \boldsymbol{B}\boldsymbol{z}_i\|_2^2$

- when $d \geq D$, sparse coding/dictionary learning

$$\min_{\boldsymbol{B} \in \mathbb{R}^{D \times d}, \boldsymbol{z}_i' s \in \mathbb{R}^d} \sum_{i=1}^m \|\boldsymbol{x}_i - \boldsymbol{B}\boldsymbol{z}_i\|_2^2 + \lambda \sum_{i=1}^m \Omega(\boldsymbol{z}_i)$$

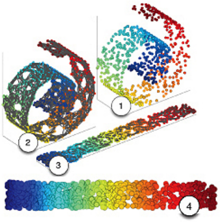
e.g., $\Omega = \|\cdot\|_1$





- Manifold, but not mathematically (i.e., differential geomety sense) rigorous
- **(No. 1?) Working hypothesis for high-dimensional data**: practical data lie (approximately) on union of low-dimensional "manifolds". Why?
  - * data generating processes often controlled by very few parameters

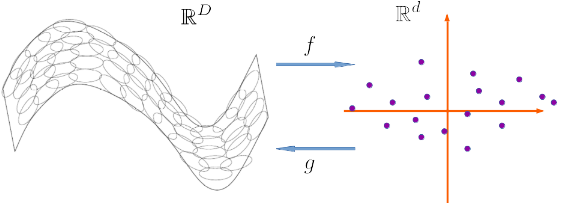
Manifold learning

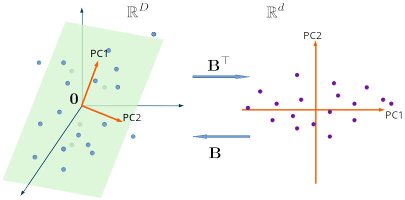


Classic methods (mostly for visualization): e.g.,

- ISOMAP [Tenenbaum, 2000]
- Locally-linear embedding [Roweis, 2000]
- Laplacian eigenmap [Belkin and Niyogi, 2001]
- t-distributed stochastic neighbor embedding (t-SNE) [van der Maaten and Hinton, 2008]

Nonlinear dimension reduction and representation learning

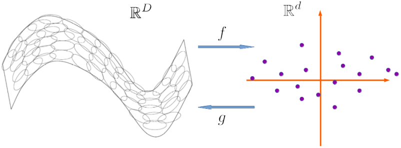




$$\min_{\boldsymbol{B}\in\mathbb{R}^{D\times d}} \sum_{i=1}^{m} \|\boldsymbol{x}_i - \boldsymbol{B}\boldsymbol{B}^\intercal \boldsymbol{x}_i\|_2^2$$

$$\min_{\boldsymbol{B}\in\mathbb{R}^{D\times d}, \boldsymbol{A}\in\mathbb{R}^{D\times d}} \sum_{i=1}^{m} \|\boldsymbol{x}_i - \boldsymbol{B}\boldsymbol{A}^\intercal \boldsymbol{x}_i\|_2^2$$

nonlinear generalization of the linear mappings:



deep autoencoders

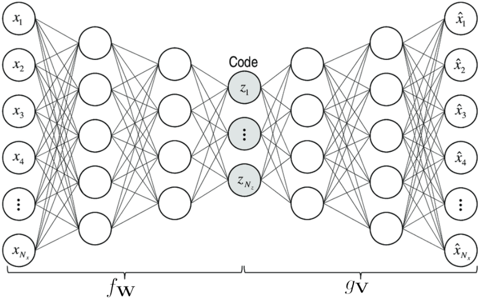
$$\min_{\boldsymbol{V},\boldsymbol{W}} \sum_{i=1}^{m} \|\boldsymbol{x}_i - g_{\boldsymbol{V}} \circ f_{\boldsymbol{W}}(\boldsymbol{x}_i)\|_2^2$$

simply $\boldsymbol{A}^\intercal \to f_{\boldsymbol{W}}$ and $\boldsymbol{B} \to g_{\boldsymbol{V}}$

A side question: why not calculate “nonlinear basis”?

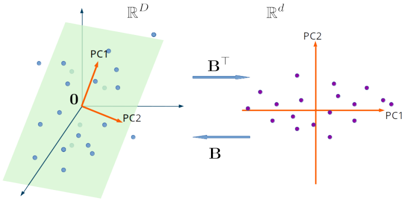
Deep autoencoders

$$\min_{\boldsymbol{V},\boldsymbol{W}} \sum_{i=1}^{m} \|\boldsymbol{x}_i - g_{\boldsymbol{V}} \circ f_{\boldsymbol{W}}(\boldsymbol{x}_i)\|_2^2$$



the landmark paper [Hinton, 2006] ... that introduced **pretraining**

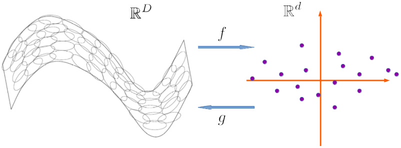
From factorization to deep factorization



factorization

$$\min_{\boldsymbol{B}\in\mathbb{R}^{D\times d},\boldsymbol{z}_i's\in\mathbb{R}^d} \sum_{i=1}^{m} \|\boldsymbol{x}_i - \boldsymbol{B}\boldsymbol{z}_i\|_2^2$$

nonlinear generalization of the linear mappings:

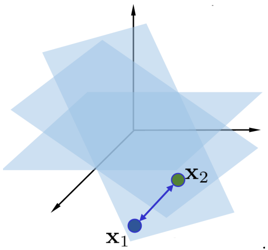


deep factorization

$$\min_{\boldsymbol{V},\boldsymbol{z}_i's\in\mathbb{R}^d} \sum_{i=1}^{m} \|\boldsymbol{x}_i - g_{\boldsymbol{V}}(\boldsymbol{z}_i)\|_2^2$$

simply $\boldsymbol{B} \to g_{\boldsymbol{V}}$

[Tan and Mayrovouniotis, 1995, Fan and Cheng, 2018, Bojanowski et al., 2017, Park et al., 2019, Li et al., 2020b], also known as **deep decoder**.



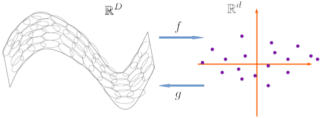
- when $d \geq D$, sparse coding/dictionary learning

$$\min_{\boldsymbol{B} \in \mathbb{R}^{D \times d}, \boldsymbol{z}_i' s \in \mathbb{R}^d} \sum_{i=1}^{m} \|\boldsymbol{x}_i - \boldsymbol{B}\boldsymbol{z}_i\|_2^2 + \lambda \sum_{i=1}^{m} \Omega\left(\boldsymbol{z}_i\right)$$

e.g., $\Omega = \|\cdot\|_1$

nonlinear generalization of the linear mappings: $(d \geq D)$

deep sparse coding/dictionary learning

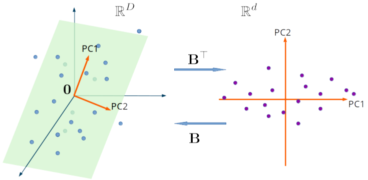


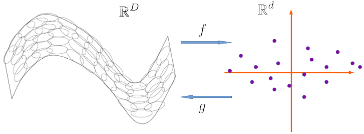
$$\min_{\boldsymbol{V}, \boldsymbol{z}_i' s \in \mathbb{R}^d} \sum_{i=1}^{m} \|\boldsymbol{x}_i - g_{\boldsymbol{V}}\left(\boldsymbol{z}_i\right)\|_2^2 + \lambda \sum_{i=1}^{m} \Omega\left(\boldsymbol{z}_i\right)$$

$$\min_{\boldsymbol{V}, \boldsymbol{W}} \sum_{i=1}^{m} \|\boldsymbol{x}_i - g_{\boldsymbol{V}} \circ f_{\boldsymbol{W}}\left(\boldsymbol{x}_i\right)\|_2^2 + \sum_{i=1}^{m} \Omega\left(f_{\boldsymbol{W}}\left(\boldsymbol{x}_i\right)\right)$$

the 2nd also called **sparse autoencoder** [Ranzato et al., 2006].

Quick summary of linear vs nonlinear models





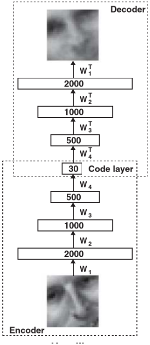
	linear models	nonlinear models
autoencoder	$\min_{\boldsymbol{B}} \sum_{i=1}^m \ell\left(\boldsymbol{x}_i, \boldsymbol{B}\boldsymbol{B}^\intercal \boldsymbol{x}_i\right)$ $\min_{\boldsymbol{B},\boldsymbol{A}} \sum_{i=1}^m \ell\left(\boldsymbol{x}_i, \boldsymbol{B}\boldsymbol{A}^\intercal \boldsymbol{x}_i\right)$	$\min_{\boldsymbol{V},\boldsymbol{W}} \sum_{i=1}^m \ell\left(\boldsymbol{x}_i, g_{\boldsymbol{V}} \circ f_{\boldsymbol{W}}\left(\boldsymbol{x}_i\right)\right)$
factorization	$\min_{\boldsymbol{B},\boldsymbol{Z}} \sum_{i=1}^m \ell\left(\boldsymbol{x}_i, \boldsymbol{B}\boldsymbol{z}_i\right)$	$\min_{\boldsymbol{V},\boldsymbol{Z}} \sum_{i=1}^m \ell\left(\boldsymbol{x}_i, g_{\boldsymbol{V}}\left(\boldsymbol{z}_i\right)\right)$
sparse coding	$\min_{\boldsymbol{B},\boldsymbol{Z}} \sum_{i=1}^m \ell\left(\boldsymbol{x}_i, \boldsymbol{B}\boldsymbol{z}_i\right)$ $+\lambda \sum_{i=1}^m \Omega\left(\boldsymbol{z}_i\right)$	$\min_{\boldsymbol{V},\boldsymbol{Z}} \sum_{i=1}^m \ell\left(\boldsymbol{x}_i, g_{\boldsymbol{V}}\left(\boldsymbol{z}_i\right)\right)$ $+\lambda \sum_{i=1}^m \Omega\left(\boldsymbol{z}_i\right)$ $\min_{\boldsymbol{V},\boldsymbol{W}} \sum_{i=1}^m \ell\left(\boldsymbol{x}_i, g_{\boldsymbol{V}} \circ f_{\boldsymbol{W}}\left(\boldsymbol{x}_i\right)\right)$ $+\lambda \sum_{i=1}^m \Omega\left(f_{\boldsymbol{W}}\left(\boldsymbol{x}_i\right)\right)$

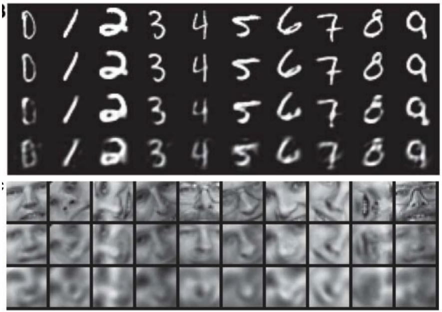
ℓ can be general loss functions other than $\|\cdot\|_2$

Ω promotes sparsity, e.g., $\Omega = \|\cdot\|_1$

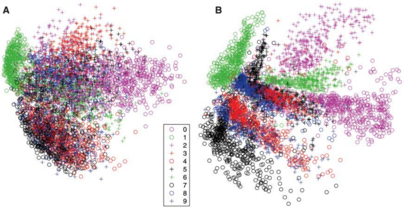
Outline

Nonlinear dimension reduction





autoencoder vs. PCA vs. logistic PCA



[Hinton, 2006]

Representation learning

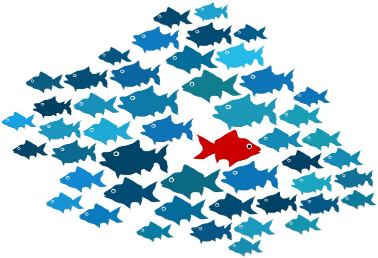
Traditional learning pipeline

- feature extraction is "independent" of the learning models and tasks
- features are handcrafted and/or learned

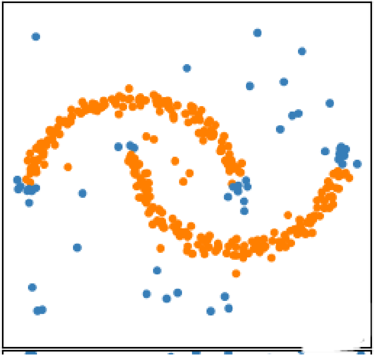
Use the low-dimensional codes as features/representations

- task agnostic
- less overfitting
- semi-supervised (rich unlabeled data + little labeled data) learning

Outlier detection



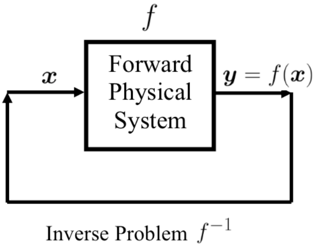
(Credit: towardsdatascience.com)



– idea: outliers don't obey the manifold assumption — the reconstruction error $\ell\left(\boldsymbol{x}_i, g_{\boldsymbol{V}} \circ f_{\boldsymbol{W}}\left(\boldsymbol{x}_i\right)\right)$ is large after autoencoder training

– for effective detection, better use ℓ that penalizes large errors less harshly than $\|\cdot\|_2^2$, e.g., $\ell\left(\boldsymbol{x}_i, g_{\boldsymbol{V}} \circ f_{\boldsymbol{W}}\left(\boldsymbol{x}_i\right)\right) = \left\|\boldsymbol{x}_i - g_{\boldsymbol{V}} \circ f_{\boldsymbol{W}}\left(\boldsymbol{x}_i\right)\right\|_2$ [Lai et al., 2019]

Deep generative prior



- **inverse problems**: given f and $\boldsymbol{y} = f(\boldsymbol{x})$, estimate $\boldsymbol{x}$
- often ill-posed, i.e., $\boldsymbol{y}$ doesn't contain enough info for recovery
- regularized data-fitting formulation:

$$\min_{\boldsymbol{x}} \ \ell(\boldsymbol{y}, f(\boldsymbol{x})) + \lambda \Omega(\boldsymbol{x})$$

where Ω contains extra info about $\boldsymbol{x}$

Suppose $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m$ come from the same manifold as $\boldsymbol{x}$

- train a deep factorization model on $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m$: $\min_{\boldsymbol{V}, \boldsymbol{Z}} \sum_{i=1}^{m} \ell(\boldsymbol{x}_i, g_{\boldsymbol{V}}(\boldsymbol{z}_i))$
- $\boldsymbol{x} \approx g_{\boldsymbol{V}}(\boldsymbol{z})$ for a certain $\boldsymbol{z}$ so: $\min_{\boldsymbol{z}} \ \ell(\boldsymbol{y}, f \circ g_{\boldsymbol{V}}(\boldsymbol{z}))$. Some recent work even uses random $\boldsymbol{V}$, i.e., without training

See: [Pan et al., 2020, Ulyanov et al., 2018, Bora et al., 2017, Wang et al., 2021, Zhuang et al., 2022]

[Bai et al., 2018] Bai, Y., Jiang, Q., and Sun, J. (2018). **Subgradient descent learns orthogonal dictionaries.** *arXiv:1810.10702*.

[Baldi and Hornik, 1989] Baldi, P. and Hornik, K. (1989). **Neural networks and principal component analysis: Learning from examples without local minima.** *Neural Networks*, 2(1):53–58.

[Belkin and Niyogi, 2001] Belkin, M. and Niyogi, P. (2001). **Laplacian eigenmaps and spectral techniques for embedding and clustering.** In Dietterich, T. G., Becker, S., and Ghahramani, Z., editors, *Advances in Neural Information Processing Systems 14 [Neural Information Processing Systems: Natural and Synthetic, NIPS 2001, December 3-8, 2001, Vancouver, British Columbia, Canada]*, pages 585–591. MIT Press.

[Bojanowski et al., 2017] Bojanowski, P., Joulin, A., Lopez-Paz, D., and Szlam, A. (2017). **Optimizing the latent space of generative networks.** *arXiv:1707.05776*.

[Bora et al., 2017] Bora, A., Jalal, A., Price, E., and Dimakis, A. G. (2017). **Compressed sensing using generative models.** In *Proceedings of the 34th International Conference on Machine Learning*, volume 70.

References ii

[Bourlard and Kamp, 1988] Bourlard, H. and Kamp, Y. (1988). **Auto-association by multilayer perceptrons and singular value decomposition.** *Biological Cybernetics*, 59(4-5):291–294.

[Elgendy, 2020] Elgendy, M. (2020). **Deep Learning for Vision Systems.** MANNING PUBN.

[Fan and Cheng, 2018] Fan, J. and Cheng, J. (2018). **Matrix completion by deep matrix factorization.** *Neural Networks*, 98:34–41.

[Hinton, 2006] Hinton, G. E. (2006). **Reducing the dimensionality of data with neural networks.** *Science*, 313(5786):504–507.

[Hinton et al., 2006] Hinton, G. E., Osindero, S., and Teh, Y.-W. (2006). **A fast learning algorithm for deep belief nets.** *Neural Computation*, 18(7):1527–1554.

[Hinton and Zemel, 1994] Hinton, G. E. and Zemel, R. S. (1994). **Autoencoders, minimum description length and helmholtz free energy.** In *Advances in neural information processing systems*, pages 3–10.

[Kawaguchi, 2016] Kawaguchi, K. (2016). **Deep learning without poor local minima.** *arXiv:1605.07110*.

References iii

[Lai et al., 2019] Lai, C.-H., Zou, D., and Lerman, G. (2019). **Robust subspace recovery layer for unsupervised anomaly detection.** *arXiv:1904.00152*.

[Li et al., 2020a] Li, S., Li, Q., Zhu, Z., Tang, G., and Wakin, M. B. (2020a). **The global geometry of centralized and distributed low-rank matrix recovery without regularization.** *arXiv:2003.10981*.

[Li et al., 2020b] Li, T., Mehta, R., Qian, Z., and Sun, J. (2020b). **Rethink autoencoders: Robust manifold learning.** *ICML workshop on Uncertainty and Robustness in Deep Learning*.

[Lu and Kawaguchi, 2017] Lu, H. and Kawaguchi, K. (2017). **Depth creates no bad local minima.** *arXvi:1702.08580*.

[Mairal, 2014] Mairal, J. (2014). **Sparse modeling for image and vision processing.** *Foundations and Trends® in Computer Graphics and Vision*, 8(2-3):85–283.

[Olshausen and Field, 1996] Olshausen, B. A. and Field, D. J. (1996). **Emergence of simple-cell receptive field properties by learning a sparse code for natural images.** *Nature*, 381(6583):607–609.

[Pan et al., 2020] Pan, X., Zhan, X., Dai, B., Lin, D., Loy, C. C., and Luo, P. (2020). **Exploiting deep generative prior for versatile image restoration and manipulation.** In *Computer Vision – ECCV 2020*, pages 262–277. Springer International Publishing.

[Park et al., 2019] Park, J. J., Florence, P., Straub, J., Newcombe, R., and Lovegrove, S. (2019). **Deepsdf: Learning continuous signed distance functions for shape representation.** In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 165–174. IEEE.

[Qu et al., 2019] Qu, Q., Zhai, Y., Li, X., Zhang, Y., and Zhu, Z. (2019). **Analysis of the optimization landscapes for overcomplete representation learning.** *arXiv:1912.02427*.

[Ranzato et al., 2006] Ranzato, M., Poultney, C. S., Chopra, S., and LeCun, Y. (2006). **Efficient learning of sparse representations with an energy-based model.** In *Advances in Neural Information Processing Systems*.

[Roweis, 2000] Roweis, S. T. (2000). **Nonlinear dimensionality reduction by locally linear embedding.** *Science*, 290(5500):2323–2326.

[Sun et al., 2017] Sun, J., Qu, Q., and Wright, J. (2017). **Complete dictionary recovery over the sphere i: Overview and the geometric picture.** *IEEE Transactions on Information Theory*, 63(2):853–884.

[Tan and Mayrovouniotis, 1995] Tan, S. and Mayrovouniotis, M. L. (1995). **Reducing data dimensionality through optimizing neural network inputs.** *AIChE Journal*, 41(6):1471–1480.

[Tenenbaum, 2000] Tenenbaum, J. B. (2000). **A global geometric framework for nonlinear dimensionality reduction.** *Science*, 290(5500):2319–2323.

[Ulyanov et al., 2018] Ulyanov, D., Vedaldi, A., and Lempitsky, V. (2018). **Deep image prior.** In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9446–9454.

[Valavi et al., 2020] Valavi, H., Liu, S., and Ramadge, P. J. (2020). **The landscape of matrix factorization revisited.** *arXiv:2002.12795*.

[van der Maaten and Hinton, 2008] van der Maaten, L. and Hinton, G. (2008). **Visualizing data using t-sne.** *Journal of Machine Learning Research*, 9:2579–2605.

[Wang et al., 2021] Wang, H., Li, T., Zhuang, Z., Chen, T., Liang, H., and Sun, J. (2021). **Early stopping for deep image prior.** *arXiv:2112.06074*.

References vi

[Zhuang et al., 2022] Zhuang, Z., Li, T., Wang, H., and Sun, J. (2022). **Blind image deblurring with unknown kernel size and substantial noise.** *arXiv:2208.09483*.