On the Theory of Deep Autoencoders

Zhouyu Shen[†] Dacheng Xiu[‡]

Chicago Booth[†]

Chicago Booth and NBER[‡]

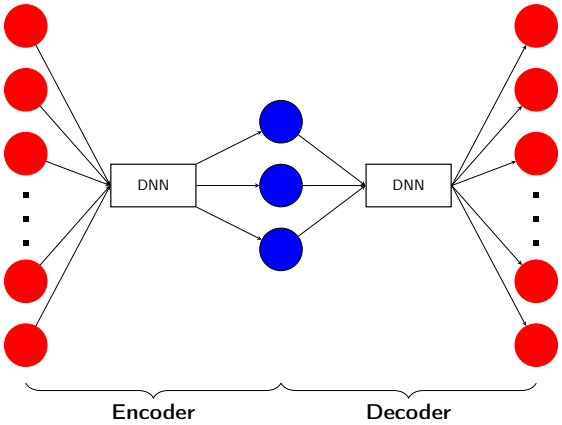
October 4, 2024

Autoencoders (AEs)

- AEs, originally proposed by LeCun (1987, PhD thesis), are specialized neural network models designed to replicate inputs at their outputs and are fundamental in unsupervised learning.

- The canonical architecture of an AE includes two key components: an encoder, which compresses the input into a lower-dimensional representation known as features, codes, embeddings, or factors, and a decoder, which reconstructs the input from this compressed form.

AEs' Canonical Architecture



Unsupervised Learning in Economics and Finance

- We are particularly drawn to AEs due to their close connection with linear factor models and their capability in conducting nonlinear dimension reduction.
- Factor Models in Economics and Finance: Stock and Watson (1999, JASA), Bai and Ng (2002, ECTA), Chamberlain and Rothschild (1983, ECTA), and Connor and Korajczyk (1986, JFE).
- The use of PCA, factor models, and matrix completion is becoming increasingly widespread, forecasting, synthetic controls, missing value interpolation, ...
- It has long been known, e.g., Baldi and Hornik (1989, Neural Network), that a single-layer AE with linear activation is equivalent to PCA.

Motivating Questions

Our paper positions AEs as estimators for nonlinear factor models, and within this framework, we address several key questions to enhance the understanding of deep and nonlinear AEs:

- Can AEs capture the "commonalities" in the inputs, a procedure often referred to as denoising, and if so, what are the statistical error bounds?
- How do AEs' architecture parameters, such as depth, width, and the number of neurons, impact their statistical performance?
- Can AEs recover the hidden low-dimensional representations in a nonlinear factor model?

Related Literature

- Nonlinear Factor Models: Etezadi-Amoli and McDonald (1983, Psychometrika), Kenny and Judd (1984, Psychol. Bull.), Amemiya and Yalcin (2001, Statist. Sci.), Griebel and Harbrecht (2014, IMA J. Numer. Anal.), Xu (2017, ICML), Agarwal et al. (2021, JASA), Freeman and Weidner (2023, JoE), Feng (2023, arxiv), Conditional AEs (Gu et al. (2021, JoE)), ...
  - Existing methods on nonlinear factor models when the underlying nonlinear functions are unknown still rely on the use of PCA

- Theory of Neural Networks: Barron (1993, TIT), Chen and White (1999, TIT), Yarotsky (2017, Neural Networks), Mei, Montanari and Nguyen (2018, PNAS), Bauer and Kohler (2019, AoS), Mei, Misiakiewicz and Montanari (2019, COLT), Schmidt-Hieber (2020, AoS), Nakada and Imaizumi (2020, JMLR), Shen et al. (2021, Neural Networks), Farrell et al. (2021, ECTA), Kohler and Langer (2021, AoS), Jiao et al. (2023, AoS) ...

AEs' Applications in CS/ML

- Feature Learning: Bourlard and Kamp (1988, Biol. Cybern.), Hinton and Zemel (1993, NIPS), Vincent et al. (2008, ICML) (Denoising AEs), Shao et al. (2017, Mech. Syst. Signal Process.),Tschannen, Bachem, and Lucic (2018, arxiv), ...
- Data Compression: Theis et al. (2017, ICLR) (Compressive AEs), Cheng et al. (2018, PCS), Habibian et al. (2019, ICCV), ...
- Noise Reduction: Lu et al. (2013, Interspeech), Xie, Xu, and Chen (2012, NIPS), Gondara (2016, ICDMW), ...
- Generative AI: Kingma and Welling (2013, ICLR) (Variational AEs), Makhzani et al. (2016, arxiv) (Adversarial AEs), ...
- Other notable models: Contractive AEs by Rifai et al. (2011, ICML), Sparse AEs by Ng (2011, Lecture Notes), Convolutional AEs by Masci et al. (2011, ICANN), Masked AEs by He et al. (2021, CVPR), ...

Nonlinear Factor Model

We examine the nonlinear factor model introduced by Yalcin and Amemiya (2001, Statist. Sci.),

$$X_{it} = X_{it}^{\star} + U_{it} = \varphi_i^{\star}(F_t^{\star}) + U_{it}, \quad i = 1, \ldots, N, \quad t = 1, \ldots, T.$$

- $F_t^{\star}$, a K-dimensional vector, represents latent factors.
- $\varphi_i^{\star} : \mathbb{R}^K \to \mathbb{R}$, an unknown function, whose functional form can vary across i.
- U_{it} accounts for idiosyncratic noise.

This framework encompasses the classical linear factor model, where $\varphi_i^{\star}(F_t^{\star}) = \Lambda_i^{\top} F_t^{\star}$, with Λ denoting loading matrix.

Boundedness Assumption

- $\text{vec}(U)$ follows the distribution characterized by $\Sigma_u^{1/2}\text{vec}(Z)$, where $Z \in \mathbb{R}^{N\times T}$ consists of independent subGaussian random variables with subGaussian norm bounded by σ_z^2. Moreover, the matrix Σ_u, which is positive semi-definite, has its spectral norm bounded.
  - This assumption holds if $U = \Sigma_1^{1/2} Z \Sigma_2^{1/2}$, where Σ_1 and Σ_2 are positive semi-definite matrices with bounded spectral norms, as is assumed by Onatski (2005, REStat) and Ahn and Horenstein (2013, ECTA).
- A constant $B > 0$ exists such that $\mathrm{P}(\sup_{1\le t\le T} \|F_t^\star\|_\infty \le B) = 1$.

Smoothness Assumption

- $\varphi_i^\star$ lies in the Hölder ball $\mathscr{H}^\beta(\Omega, B)$ with Ω an open set containing $[-B, B]^K$.
  - The Hölder ball $\mathscr{H}^\beta(\Omega, B)$ is defined as

$$\left\{ f : \Omega \to \mathbb{R}, \max_{\alpha, |\alpha| \leq \lfloor \beta \rfloor} \sup_{x \in \Omega} |D^\alpha f(x)| + \max_{\alpha : \|\alpha\|_1 = \lfloor \beta \rfloor} \sup_{\substack{x, x' \in \Omega \\ x \neq x'}} \frac{|D^\alpha f(x) - D^\alpha f(x')|}{\|x - x'\|^{\beta - \lfloor \beta \rfloor}} \leq B \right\},$$

    where $\lfloor \beta \rfloor$ represents the largest integer which is strictly smaller than β.

These assumptions are standard in the literature, ensuring that $\varphi_i^\star$ can be well approximated by neural networks.

Deep Neural Networks (DNNs)

The function f of a DNN with architecture parameters (d, w) can be expressed as:

$$f : \mathbb{R}^{n_0} \to \mathbb{R}^{n_{d+1}}, \quad \mathrm{x} \to f(\mathrm{x}) = W_d \sigma_{v_d} W_{d-1} \sigma_{v_{d-1}} \cdots W_1 \sigma_{v_1} W_0 \mathrm{x}. \tag{1}$$

- W_i represents the weight matrix and v_i the shift (bias) vector at layer i.
- d denotes depth of the DNN, whereas w is its width. n_0 and n_{d+1} represent the dimensions of the input and output.
- $\sigma_{\mathrm{x}} : \mathbb{R}^r \to \mathbb{R}^r$ is the shifted ReLU activation function:

$$\sigma_{\mathrm{x}} \begin{pmatrix} y_1 \\ \vdots \\ y_r \end{pmatrix} = \begin{pmatrix} \max(y_1 - x_1, 0) \\ \vdots \\ \max(y_r - x_r, 0) \end{pmatrix},$$

where $\mathrm{x} = (x_1, \ldots, x_r) \in \mathbb{R}^r$.

Define function class of DNNs

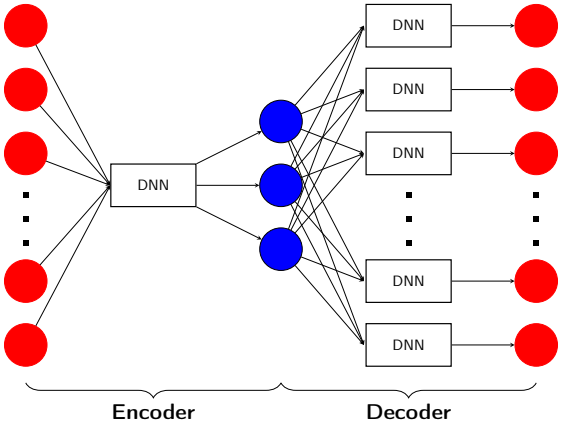
$$\mathscr{F}_{n_0}^{n_{d+1}}(d, w, C, B) := \left\{ f \text{ of the form (1)} : \|f\|_\infty \le B, \max_{j=0,\ldots,d} \left\|W_j\right\|_\infty, \max_{j=1,\ldots,d} \left\|v_j\right\|_\infty \le C \right\}.$$

Our AEs' Mathematical Formulation

We analyze a special class of AEs that have a disjoint output decoder:

- For any input $N \times 1$ vector X_t, the i-th output of this AE is given by
$$\varphi_i(\rho(X_t)), \quad i = 1, 2, \ldots, N,$$
where ρ and φ_i are DNNs.
- Formally, the AE class can be defined as follows:
$$\mathscr{F}_{\text{AE}}^{K_1} := \left\{(\rho, \varphi_1, \ldots, \varphi_N) : \rho \in \mathscr{F}_{K_0}^{K_1}(d_1, w_1, T^{5\beta+5}, B), \varphi_i \in \mathscr{F}_{K_1}^{1}(d_2, w_2, T^{5\beta+5}, B)\right\}.$$
- K_1 is the pre-selected number of neurons in the AE's bottleneck layer.

AE Architecture Featuring a Disjoint Output Decoder



Encoder: $x \to z = \rho(x)$; Decoder: $z \to (\varphi_1(z), \varphi_2(z), \ldots, \varphi_N(z))^\top$.

Comparing Disjoint-Output and Fully-Connected Decoders

AEs with disjoint output decoders can be considered a sparse-link variant of of the conventional fully connected AEs.

- This sparsity enhances network training efficiency by allowing a separate NN function, $\varphi_i(\cdot)$, to be estimated for each output X_i, thereby reducing the number of parameters that need to be learned.
  - The number of neurons is reduced by a factor of N.

Moreover, we demonstrate that

- Disjoint output decoders can effectively approximate $\varphi_i^\star(\cdot)$.
- Reducing the number of parameters helps to maintain the estimation error well under control uniformly across all is.

Training AEs

Training an AE yields a solution to the following nonlinear least squares problem:

$$(\widehat{\rho}, \widehat{\varphi}_1, \ldots, \widehat{\varphi}_N) = \underset{(W,v,\rho,\varphi_1,\ldots,\varphi_N)\in\mathscr{F}_{\mathrm{AE}}^{K_1}}{\arg\min} \sum_{t=1}^{T}\sum_{i=1}^{N} (\varphi_i \circ \rho(X_t) - X_{it})^2. \tag{2}$$

As a result,

$$\widehat{X}_{it} = \widehat{\varphi}_i \circ \widehat{\rho}(X_t).$$

- In practice, to find a desirable solution, we adopt stochastic gradient descent with adaptive learning rates (e.g., RMSprop, Adam, ...)
- The key tuning parameter is K_1, which is closely tied to the architecture of the AE.
  - A scree plot helps illuminate the required number of "linear" factors.

Error Decomposition

$$\sum_{t=1}^{T}\sum_{i=1}^{N}(\widehat{X}_{it}-\varphi_i^\star(F_t^\star))^2 \lesssim \underbrace{\sum_{t=1}^{T}\sum_{i=1}^{N}(\varphi_i^\dagger(\rho^\dagger(X_t))-\varphi_i^\star(F_t^\star))^2}_{\text{Approximation Error}} + \underbrace{\sum_{t=1}^{T}\sum_{i=1}^{N}(\varphi_i^\dagger(\rho^\dagger(X_t))-\widehat{X}_{it})^2}_{\text{Estimation Error}}.$$

- Approximation Error $\lesssim_{\mathrm{P}} NT\left(T^{-\frac{2\beta}{2\beta+K}}+T^{-1}\inf_{\rho\in\mathscr{F}_0}\sum_{t=1}^{T}\|\rho(X_t)-F_t^\star\|^2\right)\log^4(T)$,

  where $K_1 \geq K$, and $d_2 \asymp \log(T)$, $w_2 \asymp T^{\frac{K}{2(2\beta+K)}}$, and $\mathscr{F}_0 := \mathscr{F}_N^K(d_1, w_1, T^{5\beta+5}, B)$.
- Estimation Error $\lesssim_{\mathrm{P}} (TK_1 + NT^{\frac{K}{2\beta+K}})\log^4(T)$,
  - $NT^{\frac{K}{2\beta+K}}$ is effectively the total number of parameters in the decoder and TK_1 comes from the estimation error of the bottleneck layer.

AEs' Denoising Performance

- Suppose that $K \leq K_1 \leq w_0$, $d_2 \asymp \log(T)$, $w_2 \asymp T^{\frac{K}{2(2\beta+K)}}$. With probability at least $1 - C\exp(-cT)$, for $\min(N, T)$ sufficiently large,

$$\frac{1}{NT}\sum_{t=1}^{T}\sum_{i=1}^{N}(\widehat{X}_{it} - \varphi_i^\star(F_t^\star))^2 \leq \left(T^{-\frac{2\beta}{2\beta+K}} + N^{-1}K_1 + T^{-1}\inf_{\rho\in\mathscr{F}_0}\sum_{t=1}^{T}\|\rho(X_t) - F_t^\star\|^2\right)\log^4(T).$$

  - The first term in the error, $T^{-\frac{2\beta}{2\beta+K}}$, is the minimax rate as if factors are known. For a nonparametric supervised learning task, this rate can be achieved by estimators other than NNs, see, for example, Peckman (1985, AoS), Newey (1997, JoE), among others.
  - The second term comes from the estimation error of the bottleneck layer, which corroborates the results by Bai (2003, ECTA). Choosing more factors than necessary does not negatively affect the convergence rate, demonstrating the model's robustness against overestimation of factor numbers.

Overparameterized Encoder

In considering the third term, we observe that when the encoder's width and depth are overparameterized, the following holds:

$$T^{-1} \inf_{\rho \in \mathscr{F}_1} \sum_{t=1}^{T} \|\rho(X_t) - F_t^\star\|^2 = 0.$$

- In this case, the encoder $\hat{\rho}$ effectively overfits the data, achieving optimal in-sample performance.
- However, when a new data point X_{T+1} is processed through the encoder, $\hat{\varphi}(X_{T+1})$ lacks information about the factor $F_{T+1}^\star$ due to overfitting, which results in poor out-of-sample performance.

Factor Pervasiveness

With an additional assumption on factor pervasiveness, we are able to ensure the out-of-sample performance with a possibly sparse encoder.

- There exists a matrix $W^\star \in \mathbb{R}^{K\times N}$ satisfying $\|W^\star\|_\infty \le L^{-1}B$ and $\|W^\star\|_0 \asymp L$ such that for some fixed constant $c>0$, the following holds:

$$c\|x-y\| \le \|W^\star\varphi^\star(x) - W^\star\varphi^\star(y)\|, \quad \text{for any } x,y\in[-B,B]^K. \tag{3}$$

  - Intuitively, this assumption guarantees that there exist L variables containing sufficient information for estimating the factors.
  - In the context of a linear factor model, where $\varphi^\star(F_t) = \Lambda F_t^\star$, setting $W^\star = (\Lambda^\top\Lambda)^{-1}\Lambda^\top$ ensures that $W^\star\varphi^\star(x) = x$, thus satisfying inequality (3).

Properly Parameterized Encoder

- Under the aforementioned assumptions, if $K \le K_1 \le \min(w_1, w_2)$, $d_1 \asymp d_2 \asymp \log(T)$, and $w_1 \asymp w_2 \asymp T^{\frac{K}{2(2\beta+K)}}$. Additionally, we assume total number of weights in the encoder is asymptotically bounded by $L + T^{\frac{K}{2\beta+K}} \log T$ and $\log T = o(L)$, then with probability at least $1 - C\exp(-cT) - C\exp(-cL)$, as $\min(N, T)$ becomes large enough, we have:

$$\frac{1}{NT}\sum_{t=1}^{T}\sum_{i=1}^{N}\left(\hat{X}_{it} - \varphi_i^\star(F_t^\star)\right)^2 \lesssim (N^{-1}K_1 + T^{-\frac{2\beta}{2\beta+K}} + L^{-1})\log^4(T),$$

  where $\hat{X}_{it} := \hat{\varphi}_i(\hat{\rho}(X_t))$, c and C are constants independent of N, T.

- Moreover, when the data is i.i.d., for a new data X_{T+1}, we have

$$\frac{1}{N}\sum_{i=1}^{N}\mathbb{E}\left(\hat{\varphi}_i \circ \hat{\rho}(X_{T+1}) - \varphi_i^\star(F_{T+1}^\star)\right)^2 \lesssim (N^{-1}K_1 + T^{-\frac{2\beta}{2\beta+K}} + L^{-1} + T^{-1}L)\log^4(T).$$

  - As long as $\min(N, T, L, L^{-1}T) \to \infty$, the out-of-sample error converges to zero.

Feature (Factor) Learning

- Factors are identifiable up to invertible transformations.

  With any injective function $\mu : \mathbb{R}^K \to \mathbb{R}^K$, the DGP can be rewritten as:

  $$X_{it} = \varphi_i^\star \circ \mu^{-1} \circ \mu(F_t^\star) + U_{it},$$

  so that $\mu(F_t^\star)$ can, equivalently, serve as factors.

- We show that, for $\widehat{F}_t = \widehat{\rho}(X_t)$, there exists a function $\mu : \mathbb{R}^{K_1} \to \mathbb{R}^K$, composed of ρ and $\hat{\varphi}$, such that with probability at least $1 - C\exp(-cT)$, it holds that

  $$\frac{1}{T}\sum_{t=1}^{T} \|\mu(\widehat{F}_t) - F_t^\star\|_2^2 \leq C(N^{-1}K_1 + T^{-\frac{2\beta}{2\beta+K}} + T^{-1} \inf_{\rho \in \mathscr{F}_0} \sum_{t=1}^{T} \|\rho(X_t) - F_t^\star\|^2)\log^4(T).$$

Simulated DGPs

We conduct Monte Carlo simulations to assess the comparative performance of AEs and PCA based on the following DGPs:

- **Linear**: $\varphi_i^\star(F_t^\star) = C\lambda_i^\top F_t^\star$
- **Nonlinear**:
  - **Model 1**: $\varphi_i^\star(F_t^\star) = C\exp(\lambda_i^\top F_t^\star)$
  - **Model 2**: $\varphi_i^\star(F_t^\star) = C\exp(-\|\lambda_i - F_t^\star\|^2)$
  - **Model 3**: $\varphi_i^\star(F_t^\star) = C_1(\lambda_{1i}^\top F_t^\star) + C_2(\lambda_{2i}^\top F_t^\star)^2 + C_3(\lambda_{3i}^\top F_t^\star)^3$

Here we set $K = 5$, i.e., $F_t^\star \in \mathbb{R}^5$.

- Element-wise, $\lambda_i, \lambda_{1i}, \lambda_{2i}, \lambda_{3i} \overset{i.i.d.}{\sim} \mathbb{U}(-1,1)$, $F_t^\star \overset{i.i.d.}{\sim} \mathbb{U}(-2,2)$, $U_t \sim \mathcal{N}(0,1)$.
- Calibrate C, C_1, C_2, C_3 such that the variance of $\varphi_i^\star(F_t^\star)$ is 1, and for model 3, the three individual terms have equal variances.

PCA as Benchmark

PCA is traditionally associated with linear factor models, but recent research indicates its effectiveness on nonlinear factor models.

- Udell and Townsend (2019, SIAM J. Math. Data Sci.) established that large matrices with small spectral norms can be approximated by low-rank matrices.
  - Let $X \in \mathbb{R}^{m \times n}$ with $m \geq n$ and $0 < \varepsilon < 1$. Then, with $r = \lceil 72 \log(2n+1) \ / \varepsilon^2 \rceil$,

$$\inf_{\text{rank}(Y) \leq r} \|X - Y\|_\infty \leq \varepsilon \|X\|_2.$$

- In nonlinear factor models with $\varphi_i^\star(F_t^\star) = \varpi(\lambda_i, F_t^\star)$, $\lambda_i, F_t^\star \in \mathbb{R}^K$, Griebel and Harbrecht (2014, IMA J. Numer. Anal.) and Xu (2017, ICML) proved that the matrix $X_{it} := \varpi(\lambda_i, F_t^\star)$ allows for a low-rank representation:
  - For any $\delta > 0$, with $r \asymp \delta^{-K}$,

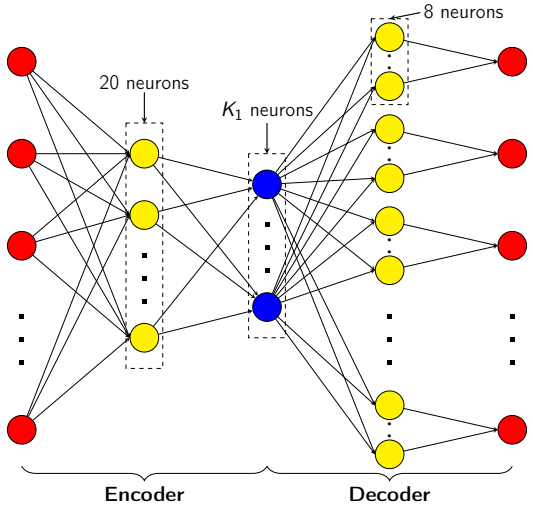
$$\inf_{\text{rank}(Y) \leq r} \|X - Y\|_\infty \lesssim \delta^\beta.$$

Simulation Details

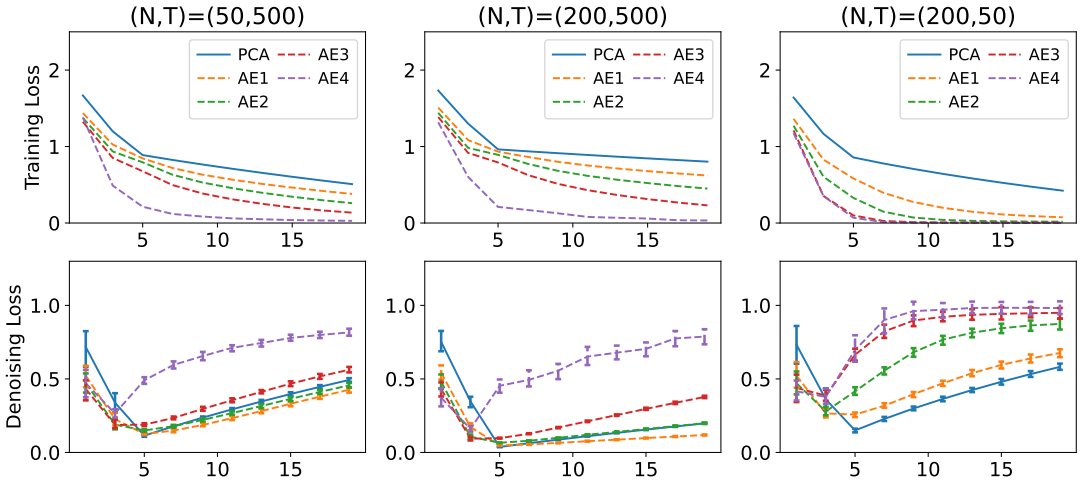
We conduct 100 simulations and report the average training loss, $N^{-1}T^{-1}\sum_{t=1}^{T}\sum_{i=1}^{N}(\widehat{X}_{it}-X_{it})^2$, and denoising loss, $N^{-1}T^{-1}\sum_{t=1}^{T}\sum_{i=1}^{N}(\widehat{X}_{it}-\varphi_i^{\star}(F_t^{\star}))^2$, where

- $(N,T)=(50,500),(200,500),(200,50)$.
- For PCA, we vary the number of factors from 1 to 20 in increments of two.
- For AE, we consider an architecture with a single hidden layer in the encoder with 20 neurons and K_1 varying from 1 to 20.
  - AE1: Single hidden layer in the decoder with two neurons.
  - AE2: Single hidden layer in the decoder with four neurons.
  - AE3: Single hidden layer in the decoder with eight neurons.
  - AE4: A fully connected decoder based on the model AE3.

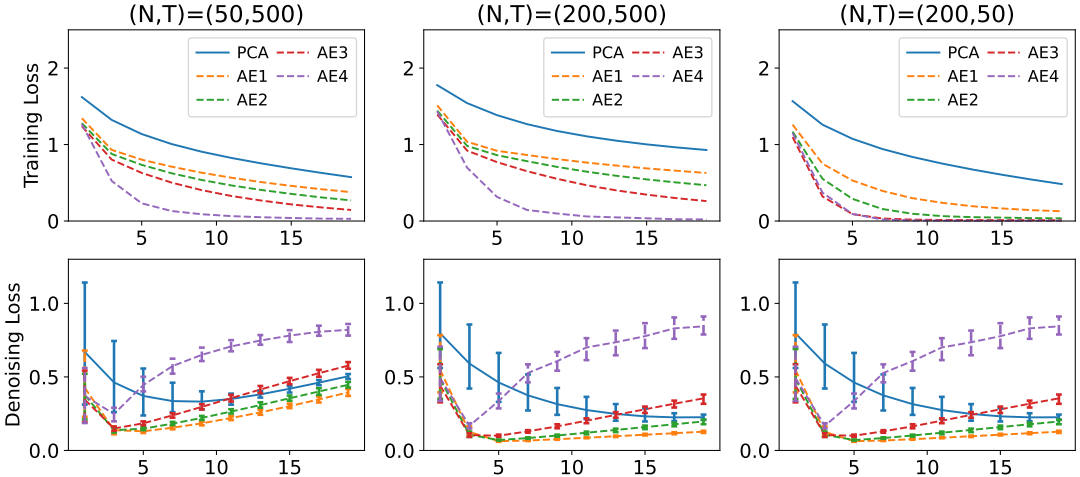
AE3's Architecture for Simulation



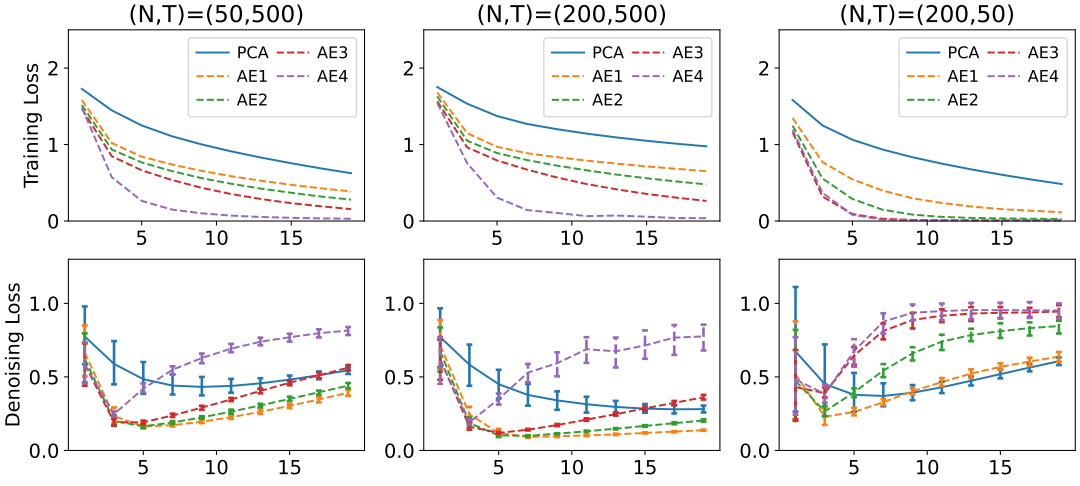
Simulation Results for Linear Model



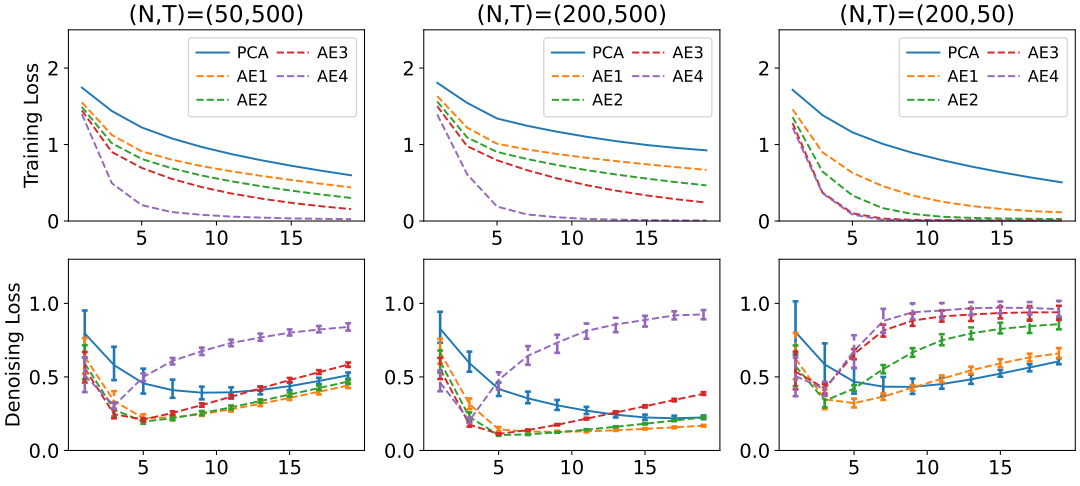
Simulation Results for Model1



Simulation Results for Model2



Simulation Results for Model3



Nowcasting GDP Growth

- Dataset: From Giannone et al. (2008, J. Monet. Econ.), focusing on predicting quarterly real GDP growth using 189 US macroeconomic indicators from January 1982 to December 2004.
- We consider a simplified exercise by using only end-of-quarter data and view the GDP to nowcast as missing values.
- Expanding window method is used. First, train AE and PCA on data from Q1 1982 to Q1 1995 and treat GDP growth in Q1 1995 as missing; output $\hat{X}_{NT}$ represents the nowcasting result for this quarter. We then adding one quarter data into the training data and repeat this exercise until Q4 2004.

Nowcasting Results: Out-of-Sample Evaluation

- Mean and standard deviation of squared error for AE, PCA, and Historical Mean(HM)'s nowcasting results are reported.

Table: Nowcasts real GDP growth: out-of-sample evaluation

	K=1	K=2	K=3	K=4	K=5
HM	4.47 (5.88)	4.47 (5.88)	4.47 (5.88)	4.47 (5.88)	4.47 (5.88)
PCA	3.49 (4.34)	3.52 (4.29)	3.50 (4.45)	3.70 (4.56)	3.54 (4.94)
AE1	**3.26** (4.03)	3.37 (4.38)	**3.41** (4.73)	**3.47** (4.77)	**3.53** (4.81)
AE2	3.33 (4.09)	**3.36** (4.29)	3.52 (4.63)	3.73 (4.89)	3.67 (4.80)
AE3	3.47 (4.06)	3.45 (4.52)	3.66 (4.80)	3.91 (5.03)	3.91 (5.07)

Nonlinear Asset Pricing Model

- According to arbitrage pricing theory, asset returns follow a linear factor model:

$$R_{it} = \lambda_i F_t^\star + U_{it}.$$

- Borri et al. (2024, arXiv) proposed that excess returns can be modeled as:

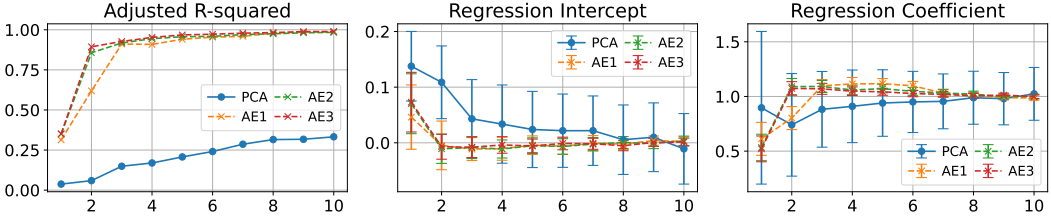
$$R_{it} = \varphi(\Lambda_i^\top F_t^\star) + U_{it},$$

where $\Lambda_i, F_t^\star \in \mathbb{R}^K$ with $K = 1$, and they fit a polynomial function for φ.

- In contrast, we employ AEs to estimate this model. This is achieved by constraining the parameters after the first layer of the decoder to be identical across all outputs and removing the bias in the first layer, ensuring that the output takes the form $\varphi(\Lambda_i^\top F_t^\star)$.
  - By using AEs, we can handle cases where $K > 1$ and estimate more complex models beyond polynomial functions.

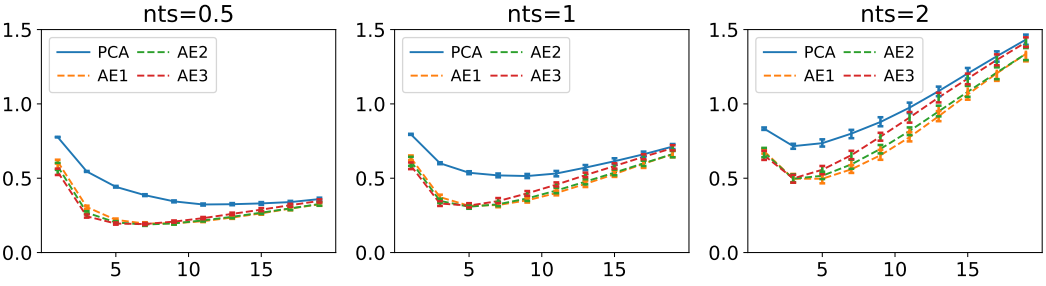
Cross-Sectional Asset Pricing

- Dataset: Long-short portfolios from Chen and Zimmermann (2022, CFR), covering the period from January 1970 to December 2022 (N = 140, T = 636).
- Using the estimator $\hat{R}_{it}$ obtained from both PCA and AEs, we conduct a cross-sectional regression by regressing the time-series average of realized excess returns on the time-series average of estimated returns: $T^{-1}\sum_{t=1}^{T} R_{it} \sim T^{-1}\sum_{t=1}^{T} \hat{R}_{it}$.
- Under the null hypothesis $\mathbb{E}[R_{it}] = \mathbb{E}[\varphi(\Lambda_i^\top F_t^\star)]$, the regression R^2 should approach one, and the regression intercept should be close to zero, indicating no mispricing.



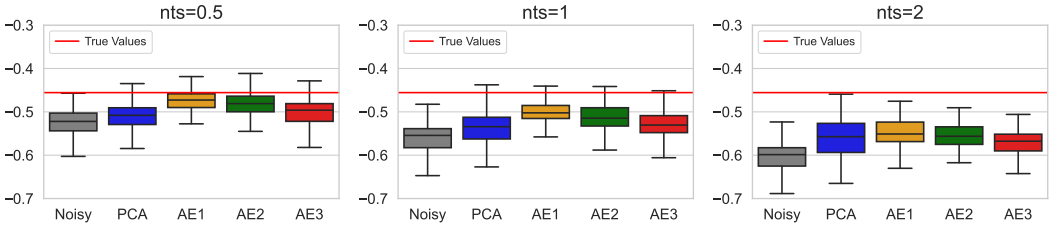
Denoising Measurement Error

- Dataset: Based on the seminal paper by Autor et al. (2013, AER). Dataset at the commuting zone (CZ) level with 722 CZs, each characterized by 30 covariates.
- We manually add Gaussian noise manually into the data X with noise-to-signal ratios of 0.5, 1, and 2, applying AE and PCA and calculating the MSE between the original data X and the denoised output $\hat{X}$.



The Impact of Import Competition on U.S. Labor Markets

- We apply 2SLS to the noisy data and denoised data obtained from both AEs and PCA with $K_1 = 5$.
  - When we apply 2SLS to the outputs of AEs and PCA with raw data, the estimated causal effects are close to the true value of -0.46. Specifically, the estimates are -0.44 for AE1, -0.44 for AE2, -0.48 for AE3, and -0.47 for PCA. These results suggest that the raw data is nearly "noiseless."
- A boxplot of the estimated causal effects from 100 repetitions is plotted.



- The red dashed line represents the causal effect derived from clean data.

Conclusion

- AEs hold promise for nonlinear dimension reduction.
- We provide non-asymptotic analysis to AEs' denoising errors and factor learning errors, which achieves the optimal rate for nonparametric regression under mild conditions.
- Through simulations and empirical illustrations, AEs provide superior performance compared with PCA under nonlinear factor models.