Complexity in Factor Pricing Models

Antoine Didisheim	Barry Ke	Bryan Kelly	Semyon Malamud
Uni. Melbourne	*Yale*	*Yale*	*EPFL*

Conditionally Efficient Portfolios Versus Managed Portfolios I

- Stock returns $R_{t+1} \in \mathbb{R}^N$
- mean-variance optimization:

$$\pi_t = \arg\max_{\pi_t} \left(E_t[\pi_t' R_{t+1}] - 0.5 \underbrace{\gamma}_{risk\ \ aversion} \mathrm{Var}_t[\pi_t' R_{t+1}] \right) \tag{1}$$

and hence the Mean-Variance Efficient (MVE) portfolio is

$$\underbrace{\pi_t}_{tangency\ \ portfolio} = \gamma^{-1} \underbrace{(\mathrm{Var}_t[R_{t+1}])^{-1}}_{N \times N\ \ covariance\ \ matrix} \underbrace{E_t[R_{t+1}]}_{N \times 1\ \ expected\ \ returns} \tag{2}$$

Conditionally Efficient Portfolios Versus Managed Portfolios II

- Similarly,

$$\tilde{\pi}_t \;=\; \gamma^{-1}(E_t[R_{t+1}R'_{t+1}])^{-1}\,E_t[R_{t+1}] \;=\; \frac{1}{1+E_t[R_{t+1}]'\,\mathrm{Var}_t[R_{t+1}]^{-1}E_t[R_{t+1}]}\pi_t \tag{3}$$

where

$$E_t[R_{t+1}R'_{t+1}] \;=\; \mathrm{Var}_t[R_{t+1}] \;+\; E_t[R_{t+1}]E_t[R_{t+1}]' \tag{4}$$

- Link to SDF:

$$E_t[R_{i,t+1}\,M_{t,t+1}] \;=\; 0 \tag{5}$$

with

$$M_{t+1} \;=\; 1 \;-\; \tilde{\pi}'_t R_{t+1} \tag{6}$$

- Now comes the big question: How do we measure the conditional expectations, $E_t[R_{t+1}]$ and $E_t[R_{t+1}R'_{t+1}]$?

Managed Portfolios and Rich Conditional Factor Structures

- Suppose $R_{i,t+1} \;=\; \underbrace{S'_{i,t}}_{conditional\;\;betas} \cdot \underbrace{\tilde{F}_{t+1}}_{latent\;\;factors} + \varepsilon_{i,t+1}$
-
$$\tilde{M}_{t+1} \;=\; 1 \;-\; W(S_t)' R_{t+1}\,, \tag{7}$$
  where
$$W(S_t) \;=\; \underbrace{(S_t \Sigma_{F,t} S'_t + \Sigma_\varepsilon)^{-1}}_{conditional\;\;covariance} \quad \underbrace{S_t \lambda_F}_{conditional\;\;expectation} \tag{8}$$
- Define **managed portfolios**
$$F_{t+1} \;=\; S'_t R_{t+1}. \tag{9}$$
  with
$$\lambda = E[F_{t+1} F'_{t+1}]^{-1} E[F_{t+1}] \tag{10}$$

Theorem

Suppose that in the limit, as $P \to \infty$, *the vector of latent risk premia* λ_F *satisfies*

$$\lambda_F' A \lambda_F \;\to\; 0 \tag{11}$$

for any symmetric, positive definite A *with uniformly bounded trace. Let*

$$M_{t+1} = 1 - \lambda' F_{t+1}, \tag{12}$$

be the factor approximation for the SDF with λ *given by* (10). *Then,* M_{t+1} *converges to* $\tilde{M}_{t+1}$ *and the Sharpe ratio of* $\lambda' F_{t+1}$ *converges to that of* $W(S_t)' R_{t+1}$ *as* $P \to \infty$.

This Paper: ML in Cross-sectional Asset Pricing

- Main theoretical result: SDF performance generally *increasing* in model complexity
  - Higher portfolio Sharpe ratio
  - Smaller pricing errors
- Prior evidence of empirical gains from ML are *what we should expect*
- **Direct empirical support for theory**

Complexity in the Cross Section: A Brief History

SDF representable as managed portfolios $M^{\star}_{t+1} = 1 - \sum_{i=1}^{n} w(X_t)' R_{i,t+1}$, s.t. $E_t[M^{\star}_{t+1} R_{i,t+1}] = 0 \ \forall i$

- Cross-sectional asset pricing is about $w_t = w(X_t)$
  - Explains differences in average returns
  - Defines the MVE portfolio

Complexity in the Cross Section: A Brief History

$$\boxed{\text{SDF representable as managed portfolios } M^\star_{t+1} = 1 - \sum_{i=1}^n w(X_t)' R_{i,t+1}, \text{ s.t. } E_t[M^\star_{t+1} R_{i,t+1}] = 0 \ \forall i}$$

- Cross-sectional asset pricing is about $w_t = w(X_t)$
  - Explains differences in average returns
  - Defines the MVE portfolio
- Why does cross-section literature rarely start here? Because w must be estimated
  - This is a high-dimensional (*complex*) problem
  - We know: In-sample tangency portfolio behaves horribly out-of-sample
  - Why? Complexity ($n/T \not\to 0$) $\rightarrow$ LLN doesn't apply $\rightarrow$ IS and OOS diverge

Complexity in the Cross Section: A Brief History

$\boxed{\text{SDF representable as managed portfolios } M^{\star}_{t+1} = 1 - \sum_{i=1}^{n} w(X_t)' R_{i,t+1}, \text{ s.t. } E_t[M^{\star}_{t+1} R_{i,t+1}] = 0 \ \forall i}$

- Cross-sectional asset pricing is about $w_t = w(X_t)$
  - Explains differences in average returns
  - Defines the MVE portfolio
- Why does cross-section literature rarely start here? Because w must be estimated
  - This is a high-dimensional (*complex*) problem
  - We know: In-sample tangency portfolio behaves horribly out-of-sample
  - Why? Complexity ($n/T \not\to 0$) $\to$ LLN doesn't apply $\to$ IS and OOS diverge
- Standard solution: Restrict w
  - E.g., Fama-French: $w_{i,t} = b_0 + b_1 \text{Size}_{i,t} + b_2 \text{Value}_{i,t}$ (Brandt et al. 2007 generalize)
  - Reduces parameters, implies factor model: $M_{t+1} = 1 - b_0 MKT - b_1 SMB - b_2 HML$
  - "Shrinking the cross-section" Kozak et al. (2020) — use a few PCs of anomaly factors

Complexity in the Cross Section: The Meta-Learning Problem

- **Given a finite history** $\Theta = \{X_{i,t},\ R_{i,t+1},\ t = 1, \cdots, T\}$**, choose a learning algorithm** $\Theta \to\ G(X; \Theta)$ **such that** $\sum_{i=1}^{n} G(X_{i,T}; \Theta)' R_{i,T+1}$ **has a high Sharpe Ratio out of sample.**
- Human algorithms seem to always be **parametric families** $\hat{w}(X, \lambda),\ \lambda \in \mathbb{R}^P$, so that the learned model is $\hat{w}(X, \lambda(\Theta))$
- **the weight (parameter) vector** λ **is usually trained no minimize some form of objective in sample:**

$$\lambda \ = \ \arg\max \left\{ \textit{Realized In Sample Sharpe Ratio}\Big(\sum_{i=1}^{n} \hat{w}(X_{i,t}, \lambda)' R_{i,t+1}\Big)_{t=1}^{T} \ + \ \textit{penalty}(\lambda) \right\} \tag{13}$$

and this argmin generates $\lambda(\Theta)$

Big Question: How to Select a Good Family $\hat{w}(X, \lambda)$

- **Conventional wisdom:**
  - Realized In Sample Sharpe Ratio $\approx$ Expected Out Of Sample Sharpe Ratio
  - As long as $w(X, \lambda)$ is rich enough, its details do not matter: **universal approximation property**
  - One should try to keep $P = \dim(\lambda)$ low **to avoid overfit**

Complex Families $\hat{w}(X, \lambda)$

- **Unconventional wisdom (Complexity):**
  - Families $\hat{w}(X, \lambda)$ with $P >> T$ work great out of sample
  - Realized In Sample Sharpe Ratio $>>$ Expected Out Of Sample Sharpe Ratio and hence cannot be used directly
  - Exact details of $\hat{w}(X, \lambda)$ matter a lot because **it defines how quickly we can learn a good approximation of w in finite samples**
  - Generically, **despite the universal approximation property,** approximation breaks down: when $P/T > 0$, we have $\hat{\mathbf{w}}(\mathbf{X}, \lambda) \not\approx \mathbf{w}(\mathbf{X})$

Complexity in the Cross Section: Machine Learning Perspective

$\boxed{\text{SDF representable as } M^{\star}_{t+1} = 1 - \sum_{i=1}^{n} w(X_t)' R_{i,t+1}, \text{ s.t. } E_t[M^{\star}_{t+1} R_{i,t+1}] = 0 \ \forall i}$

Rather than restricting $w(X_t)$....

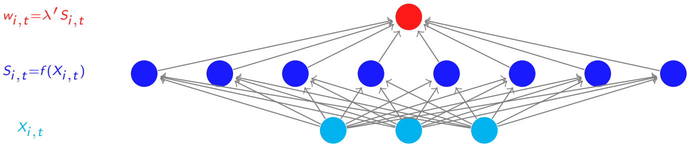
- ...expand parameterization, saturate with conditioning information

Complexity in the Cross Section: Machine Learning Perspective

$$\boxed{\text{SDF representable as } M^{\star}_{t+1} = 1 - \sum_{i=1}^{n} w(X_t)' R_{i,t+1}, \text{ s.t. } E_t[M^{\star}_{t+1} R_{i,t+1}] = 0 \ \forall i}$$

Rather than restricting $w(X_t)$....

- ...expand parameterization, saturate with conditioning information
- Approximate w with neural network: $\hat{w}(X_{i,t}, \lambda) \approx \lambda' S_{i,t}$ with a linear family
- $P \times 1$ vector $S_{i,t}$ is known nonlinear function of original predictors $\pi_{i,t}$

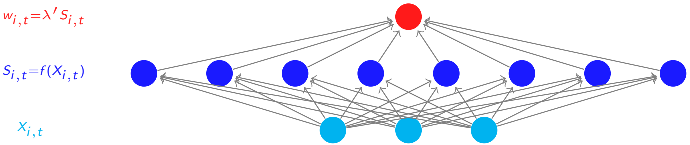


Complexity in the Cross Section: Machine Learning Perspective

$$\boxed{\text{SDF representable as } M^\star_{t+1} = 1 - \textstyle\sum_{i=1}^n w(X_t)' R_{i,t+1}, \text{ s.t. } E_t[M^\star_{t+1} R_{i,t+1}] = 0 \ \forall i}$$

Rather than restricting $w(X_t)$....

- ...expand parameterization, saturate with conditioning information
- Approximate w with neural network: $\hat{w}(X_{i,t}, \lambda) \approx \lambda' S_{i,t}$ with a linear family
- $P \times 1$ vector $S_{i,t}$ is known nonlinear function of original predictors $\pi_{i,t}$



- Implies that empirical SDF is a high-dimensional factor model with factors F_{t+1} :

$$\begin{aligned} M^\star_{t+1} \approx M_{t+1} &= 1 - \lambda' S_t' R_{t+1} \\ &= 1 - \sum_i (\lambda' S_{i,t} R_{i,t+1}) = 1 - \lambda' \underbrace{\sum_i S_{i,t} R_{i,t+1}}_{= F_{t+1} \in \mathbb{R}^{P \times 1}} = 1 - \lambda' F_{t+1} \end{aligned} \tag{14}$$

Complexity in the Cross Section: Machine Learning Perspective

True SDF: $M^{\star}_{t+1} = 1 - w(X_t)' R_{t+1}$ Empirical Model: $M_{t+1} = 1 - \underbrace{\lambda' F_{t+1}}_{P \text{ params}}$

The Objective:

- Maximize out-of-sample Sharpe ratio (equivalently, minimize out-of-sample pricing errors) of SDF

Complexity in the Cross Section: Machine Learning Perspective

True SDF: $M^{\star}_{t+1} = 1 - w(X_t)' R_{t+1}$ Empirical Model: $M_{t+1} = 1 - \underbrace{\lambda' F_{t+1}}_{P \text{ params}}$

The Objective:

- Maximize out-of-sample Sharpe ratio (equivalently, minimize out-of-sample pricing errors) of SDF

The Choice:

- Fix T data points. Decide on "complexity" (number of factors P) to use in approximating model

Complexity in the Cross Section: Machine Learning Perspective

True SDF: $M^{\star}_{t+1} = 1 - w(X_t)' R_{t+1}$ Empirical Model: $M_{t+1} = 1 - \underbrace{\lambda' F_{t+1}}_{P \text{ params}}$

The Objective:

- Maximize out-of-sample Sharpe ratio (equivalently, minimize out-of-sample pricing errors) of SDF

The Choice:

- Fix T data points. Decide on "complexity" (number of factors P) to use in approximating model

The Tradeoff:

- Simple SDF ($P << T$) has low variance (thanks to parsimony) but is a poor approximator of w
- Complex SDF ($P > T$) is good approximator but may behave poorly (and requires shrinkage)

Complexity in the Cross Section: Machine Learning Perspective

True SDF: $M^{\star}_{t+1} = 1 - w(X_t)' R_{t+1}$ Empirical Model: $M_{t+1} = 1 - \underbrace{\lambda' F_{t+1}}_{P \text{ params}}$

The Objective:

- Maximize out-of-sample Sharpe ratio (equivalently, minimize out-of-sample pricing errors) of SDF

The Choice:

- Fix T data points. Decide on "complexity" (number of factors P) to use in approximating model

The Tradeoff:

- Simple SDF ($P << T$) has low variance (thanks to parsimony) but is a poor approximator of w
- Complex SDF ($P > T$) is good approximator but may behave poorly (and requires shrinkage)

The Central Research Question:

- Which P should the analyst opt for? Does the benefit of more factors justify their cost?

Complexity in the Cross Section: Machine Learning Perspective

True SDF: $M^{\star}_{t+1} = 1 - w(X_t)' R_{t+1}$ Empirical Model: $M_{t+1} = 1 - \underbrace{\lambda' F_{t+1}}_{P \text{ params}}$

The Objective:

- Maximize out-of-sample Sharpe ratio (equivalently, minimize out-of-sample pricing errors) of SDF

The Choice:

- Fix T data points. Decide on "complexity" (number of factors P) to use in approximating model

The Tradeoff:

- Simple SDF ($P << T$) has low variance (thanks to parsimony) but is a poor approximator of w
- Complex SDF ($P > T$) is good approximator but may behave poorly (and requires shrinkage)

The Central Research Question:

- Which P should the analyst opt for? Does the benefit of more factors justify their cost?

Answer:

- Use the largest factor model (largest P) that you can compute

Theory Environment

Model

- n assets with returns R_{t+1}
- Empirical SDF $M_{t+1} = 1 - \lambda' S_t' R_{t+1}$
  - Think of S_t as "generated features" in neural net with input X_t
  - $P \times 1$ vector of instruments, S_t (i.e., P factors F_{t+1})
- (Ridge-penalized) objective

Max Sharpe Ratio		Min Pricing Error (HJ-distance)
$\min_\lambda E[(1 - \lambda' S_t' R_{t+1})^2] + z\lambda'\lambda$	or	$\min_\lambda E[MF]' E[FF']^{-1} E[MF] + z\lambda'\lambda$

Solution:

$$\hat{\lambda}(z) = \left(zI + \frac{1}{T}\sum_t F_t F_t'\right)^{-1} \frac{1}{T}\sum_t F_t \not\approx \ (\mathbf{zI} + \mathbf{E}[\mathbf{FF}'])^{-1}\mathbf{E}[\mathbf{F}]$$

Theory Environment

Model

- n assets with returns R_{t+1}
- Empirical SDF $M_{t+1} = 1 - \lambda' S_t' R_{t+1}$
  - Think of S_t as "generated features" in neural net with input X_t
  - $P \times 1$ vector of instruments, S_t (i.e., P factors F_{t+1})
- (Ridge-penalized) objective

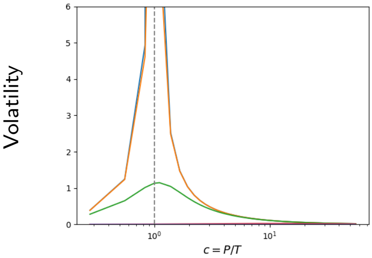
Max Sharpe Ratio		Min Pricing Error (HJ-distance)
$\min_\lambda E[(1-\lambda' S_t' R_{t+1})^2] + z\lambda'\lambda$	or	$\min_\lambda E[MF]' E[FF']^{-1} E[MF] + z\lambda'\lambda$

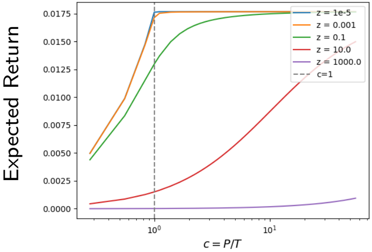
Solution:

$$\hat{\lambda}(z) = \left(zI + \frac{1}{T}\sum_t F_t F_t'\right)^{-1} \frac{1}{T}\sum_t F_t \not\approx \ (\mathbf{zI} + \mathbf{E}[\mathbf{FF}'])^{-1}\mathbf{E}[\mathbf{F}]$$

- Goal: Characterize **out-of-sample** behaviors, contrast **simple** (small P) models vs. **complex** models
- Tools: Joint limits as numbers of observations and parameters are large, $T, P \to \infty$, RMT

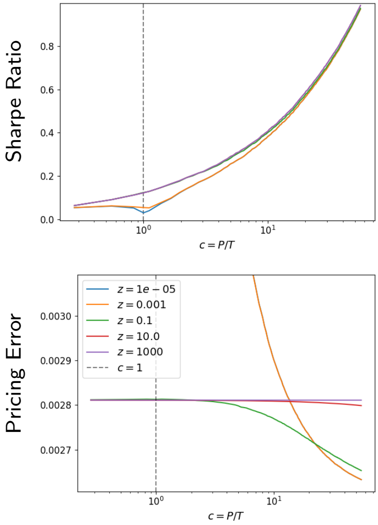
Complexity and the SDF





1. SDF variance
   - As $c \to 1$, λ variance blows up. A unique λ produces max SR, but it has a high variance
   - When $c > 1$, variance *drops* with model complexity! Why?
   - Many λ's exactly fit training data, ridge selects one with a small variance
2. SDF expected returns
   - Low for $c \approx 0$ due to poor approximation of the true model
   - Monotonically increases with model complexity

Complexity and the SDF

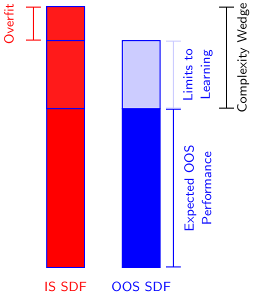


Main theory result

- Complexity is a virtue—biggest model wins
  - Approximation benefits dominate costs of heavy parameterization
  - For moderate complexity ($c \approx 1$), ridge shrinkage is beneficial
  - For high complexity ($c >> 1$), ridge shrinkage has small benefit (the important shrinkage is implicit)
- Paper provides general, rigorous theoretical statements and proofs that underlie plots
- Plots calculated from our theorems in a reasonable calibration

Complexity and the SDF: Other Theoretical Results

1. "Complexity wedge" $=$ IS Performance – Expected OOS Performance
   $= \underbrace{\text{IS} - \text{True}}_{\text{“Overfit”}} + \underbrace{\text{True} - \text{OOS}}_{\text{“Limits to Learning”}}$
   - Quantifiable based on training data
   - Can infer performance of true SDF and how far you are from it, but cannot recover it!



2. Show how to infer optimal shrinkage, z^*, from training data
3. There is no low-rank rotation of complex factors that preserves model performance (cf. Kozak, Nagel, and Santosh, 2020)

Empirical Analysis

- Analyze empirical analogs to theoretical comparative statics
- Study conventional setting with conventional data
  - Forecast target is a monthly return of US stocks from CRSP 1963–2021
  - Conditioning info ($X_{i,t}$) is 130 stock characteristics from Jensen, Kelly, and Pedersen (2022)
- Out-of-sample performance metrics are:
  - SDF Sharpe ratio
  - Mean squared pricing errors (factors as test assets)

Empirical Analysis

Random Fourier Features

- Empirical model: $M_{t+1} = 1 - \lambda' S_t' R_{t+1}$
- Need framework to smoothly transition from low to high complexity

Empirical Analysis

Random Fourier Features

- Empirical model: $M_{t+1} = 1 - \lambda' S_t' R_{t+1}$
- Need framework to smoothly transition from low to high complexity
- Adopt ML method known as "random Fourier features" (RFF)
  - Let $X_{i,t}$ be 130×1 predictors. RFF converts $X_{i,t}$ into

$$S_{\ell,i,t} = \sin(\gamma_\ell' X_{i,t}), \quad \gamma_\ell \sim iidN(0, \gamma I)$$

  - $S_{\ell,i,t}$: Random lin-combo of $X_{i,t}$ fed through non-linear activation

Empirical Analysis

Random Fourier Features

- Empirical model: $M_{t+1} = 1 - \lambda' S_t' R_{t+1}$
- Need framework to smoothly transition from low to high complexity
- Adopt ML method known as "random Fourier features" (RFF)
  - Let $X_{i,t}$ be 130×1 predictors. RFF converts $X_{i,t}$ into

$$S_{\ell,i,t} = \sin(\gamma_\ell' X_{i,t}), \quad \gamma_\ell \sim iidN(0, \gamma I)$$

  - $S_{\ell,i,t}$: Random lin-combo of $X_{i,t}$ fed through non-linear activation
- For fixed inputs can create an arbitrarily large (or small) feature set
  - Low-dim model (say $P = 1$) draw a single random weight
  - High-dim model (say $P =$ 10,000) draw many weights

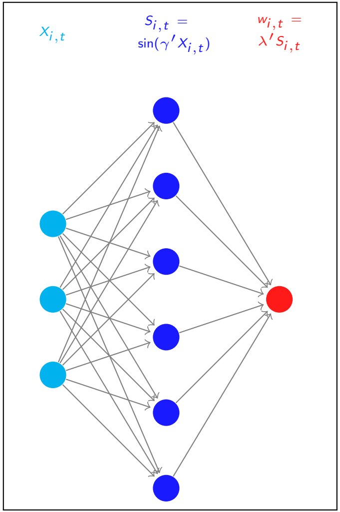
Empirical Analysis

Random Fourier Features

- Empirical model: $M_{t+1} = 1 - \lambda' S_t' R_{t+1}$
- Need framework to smoothly transition from low to high complexity
- Adopt ML method known as "random Fourier features" (RFF)
  - Let $X_{i,t}$ be 130×1 predictors. RFF converts $X_{i,t}$ into

$$S_{\ell,i,t} = \sin(\gamma_\ell' X_{i,t}), \quad \gamma_\ell \sim iidN(0, \gamma I)$$

  - $S_{\ell,i,t}$: Random lin-combo of $X_{i,t}$ fed through non-linear activation
- For fixed inputs can create an arbitrarily large (or small) feature set
  - Low-dim model (say $P = 1$) draw a single random weight
  - High-dim model (say $P = 10{,}000$) draw many weights
- In fact, RFF is a two-layer neural network with fixed weights (γ) in the first layer and optimized weights (λ) in the second layer

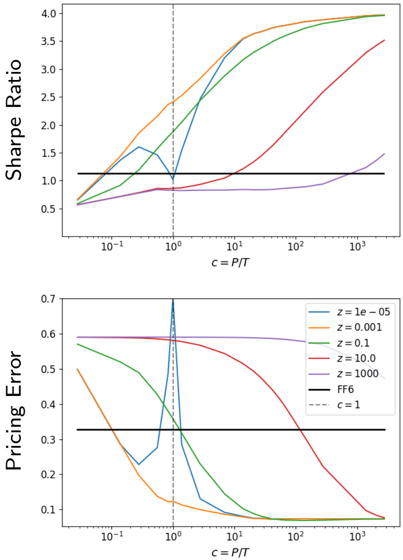


Empirical Analysis

Training and Testing

- We estimate out-of-sample SDF with:
  1. Thirty-year rolling training window ($T = 360$)
  2. Various shrinkage levels, $\log_{10}(z) = -12, \ldots, 3$
  3. Various complexity levels $P = 10^2, \ldots, 10^6$
- For each level of complexity $c = P/T$, we plot
  1. Out-of-sample Sharpe ratio of the kernels and
  2. Pricing errors on 10^6 "complex" factors: $F_{t+1} = S_t' R_{t+1}$
- Also report Sharpe ratio and pricing errors of FF6 to benchmark our results

Out-of-sample SDF Performance

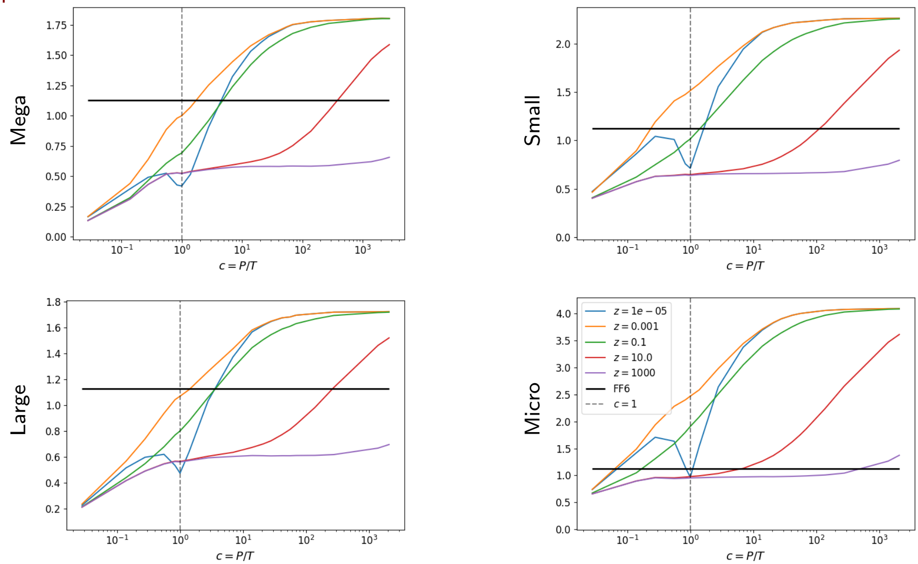


Main Empirical Result

- OOS behavior of ML-based SDF closely matches theory
- High complexity models
  - Improve over simple models by a factor of 3 or more
  - Dominate popular benchmarks like FF6

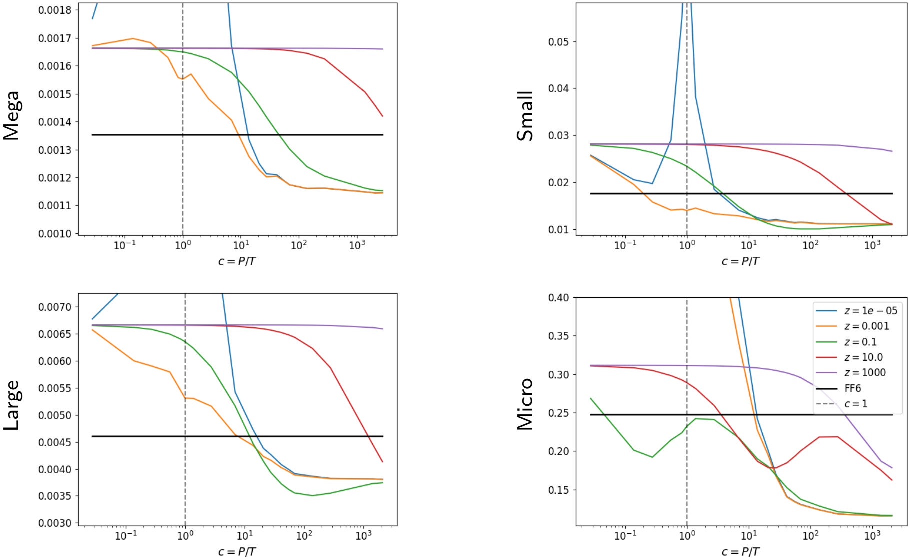
SDF Performance in Restricted Samples: Sharpe Ratio

Market Capitalization Subsamples

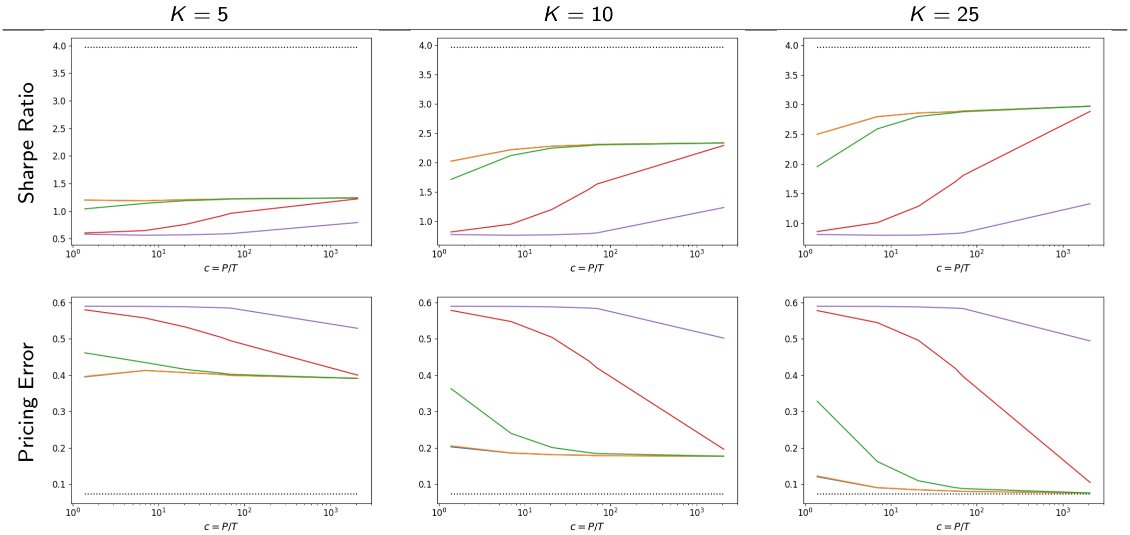


SDF Performance in Restricted Samples: Pricing Errors

Market Capitalization Subsamples



What About "Shrinking" With PCA?



Conclusions, I

- Asset pricing and asset management in midst of boom in ML research
- We provide new, rigorous theoretical insight into the behavior of ML models/portfolios
- Contrary to conventional wisdom: Higher complexity improves model performance

Virtue of Complexity: Performance of ML portfolios can be improved by pushing model parameterization far beyond the number of training observations

Conclusions, I

- Asset pricing and asset management in midst of boom in ML research
- We provide new, rigorous theoretical insight into the behavior of ML models/portfolios
- Contrary to conventional wisdom: Higher complexity improves model performance

Virtue of Complexity: Performance of ML portfolios can be improved by pushing model parameterization far beyond the number of training observations

- *Not* license to add arbitrary predictors to model. Instead, we recommend
  - i. including all plausibly relevant predictors
  - ii. using rich non-linear models rather than simple linear specifications
  - Doing so confers prediction/portfolio benefits, even when training data is scarce and particularly when accompanied by shrinkage
- In canonical empirical problem—pricing the cross section of returns—we find
  - OOS Sharpe rise by factor of 4 relative to FF6 model, pricing errors reduced by a factor of 3

Conclusions, II

- Clashes with philosophy of parsimony frequently espoused by economists
- Two oft-repeated quotes from famed statistician George Box:

All models are wrong, but some are useful.

Since all models are wrong the scientist cannot obtain a 'correct' one by excessive elaboration. On the contrary, following William of Occam, he should seek an economical description of natural phenomena. Just as the ability to devise simple but evocative models is the signature of the great scientist so overelaboration and overparameterization is often the mark of mediocrity.

Conclusions, II

- Clashes with philosophy of parsimony frequently espoused by economists
- Two oft-repeated quotes from famed statistician George Box:

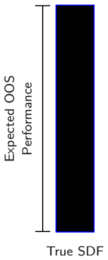
All models are wrong, but some are useful.

Since all models are wrong the scientist cannot obtain a 'correct' one by excessive elaboration. On the contrary, following William of Occam, he should seek an economical description of natural phenomena. Just as the ability to devise simple but evocative models is the signature of the great scientist so overelaboration and overparameterization is often the mark of mediocrity.

Occam's Blunder? Small model is preferable only if it is correctly specified. But models are never correctly specified. Logical conclusion?

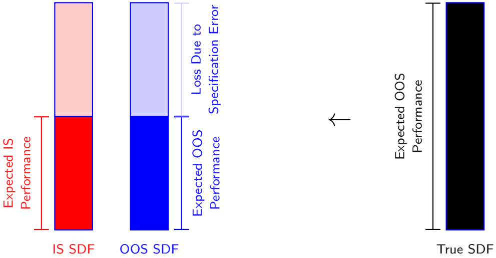
Appendix

Complexity in the Cross Section: Machine Learning Perspective



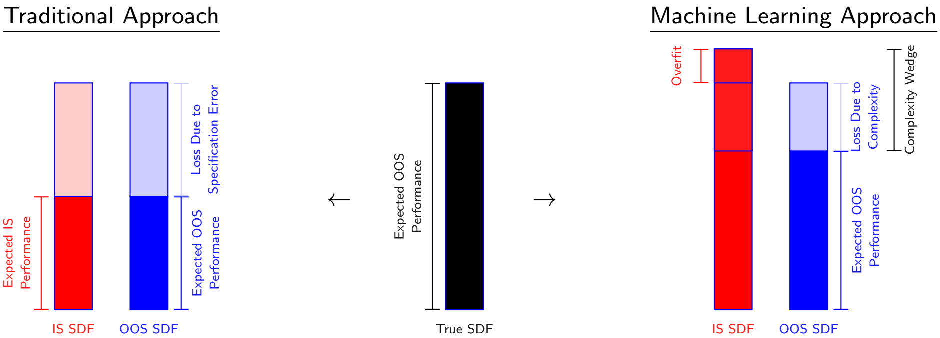
Complexity in the Cross Section: Machine Learning Perspective

Traditional Approach



- Restrict specification so $P/T \approx 0$
- Aligns IS and OOS performance
- May get lucky with spec, but can't be lucky on average
- Like shrinking *before seeing data*

Complexity in the Cross Section: Machine Learning Perspective



- Restrict specification so $P/T \approx 0$
- Aligns IS and OOS performance
- May get lucky with spec, but can't be lucky on average
- Like shrinking *before seeing data*

- $P/T \to \infty$ eliminates specification error
- IS overfit *improves* OOS performance
- Loss due to limits on learning (breakdown of LLN, high variance)
- Mitigate with shrinkage *after seeing data*

Understanding The Theory

- Suppose $c = P/T \approx 0$. Then, we know

$$\lambda \;=\; E[FF']^{-1}E[F] \;=\; \frac{1}{1+MaxSR^2}\,\mathrm{Var}[F]^{-1}E[F]\,, \tag{15}$$

where we have defined

$$MaxSR^2 \;=\; E[F]'\,\mathrm{Var}[F]^{-1}E[F] \tag{16}$$

$$E[\lambda' F_{t+1}] \;=\; E[(\lambda' F_{t+1})^2] \;=\; E[F]'E[FF']^{-1}E[F] \;=\; \frac{MaxSR^2}{1+MaxSR^2} \tag{17}$$

Principal Components and Ridge I

- $\text{Var}[F] = U \,\text{diag}(\mu) U'$, and we can define PC_i to be the i-th column of $U'F$; and

$$\theta = U'E[F] \tag{18}$$

-

$$R(PC_i) = PC_i' F_{t+1}$$

$$E[R(PC_i)] = \theta_i, \ \text{Var}[R(PC_i)] = \mu_i, \ (SR(PC_i))^2 = \frac{\theta_i^2}{\mu_i}$$

and

$$\begin{aligned} MaxSR^2 &= E[F]' \,\text{Var}[FF']^{-1} E[F] = E[F]' U \,\text{diag}(\mu^{-1}) U' E[F] \\ &= \theta' \,\text{diag}(\mu^{-1}) \theta = \sum_i \frac{\theta_i^2}{\mu_i} = \sum_i (SR(PC_i))^2 . \end{aligned} \tag{19}$$

Principal Components and Ridge II

- Define
$$\lambda(z) \;=\; (zI + E[FF'])^{-1}E[F] \tag{20}$$
and
$$R^{infeasible}(z) \;=\; F'_{t+1}\lambda(z) \tag{21}$$
- The first moment is
$$\mathcal{R}_1^{infeas}(z) \;=\; E[R^{infeasible}(z)] \;=\; E[F]'(zI + E[FF'])^{-1}E[F] \;=\; \frac{A(z)}{1+A(z)} \tag{22}$$
where
$$A(z) \;=\; E[F]'(zI + \mathrm{Var}[F])^{-1}E[F] \;=\; \sum_i (SR(PC_i))^2 \frac{\mu_i}{\mu_i + z}\,. \tag{23}$$
- and
$$\mathcal{R}_2^{infeas}(z) \;=\; E[(R^{infeasible}(z))^2] \;=\; \frac{d}{dz}\left(\frac{zA(z)}{1+A(z)}\right)\,. \tag{24}$$

Principal Components and Ridge III

In this case,

$$SR^{infeas}(z) \;=\; \frac{\mathcal{R}_1^{infeas}(z)}{(\mathcal{R}_2^{infeas}(z))^{1/2}} \tag{25}$$

is monotone decreasing in z.

Random Matrix Theory and Implicit Regularization I

- When $c = P/T > 0$, estimating $E[FF']$ and $E[F]$ becomes infeasible and

$$\hat{\lambda}(z) = \left(zI + \frac{1}{T}\sum_t F_t F_t'\right)^{-1} \frac{1}{T}\sum_t F_{t+1} \not\approx (zI + E[FF'])^{-1}E[F] \tag{26}$$

because

$$B_T = \frac{1}{T}\sum_t F_t F_t' \not\approx E[FF'] \text{ and } \bar{F}_T = \frac{1}{T}\sum_t F_{t+1} \not\approx E[F] \tag{27}$$

- Stieltjes transforms

$$\begin{aligned} m(-z) &= P^{-1}\operatorname{tr}((zI + \mathrm{Var}[FF'])^{-1}) = P^{-1}\sum_i (z + \mu_i)^{-1} \\ m(-z;c) &= P^{-1}\operatorname{tr}\left((zI + B_T)^{-1}\right) \end{aligned} \tag{28}$$

Random Matrix Theory and Implicit Regularization II

- $$\xi(z;c) \;=\; \frac{1}{T} F_{T+1}' \left(zI + B_T\right)^{-1} F_{T+1} \;\leq\; c\, z^{-1} \tag{29}$$

- The implicit shrinkage function

$$Z_*(z;c) \;=\; z\left(1 + \xi(z;c)\right) \tag{30}$$

- **Theorem** When $P \to \infty,\; P/T \to c:$

$$m(-z;c) \;=\; \frac{Z_*(z;c)}{z}\, m(-Z_*(z;c)) \tag{31}$$

Implicit Regularization and Expected Return

Recall that

$$\mathcal{R}_1^{infeas}(z) \;=\; E[R^{infeasible}(z)] \;=\; E[F]'(zI + E[FF'])^{-1}E[F] \;=\; \frac{A(z)}{1+A(z)} \tag{32}$$

Our goal is to understand

$$\mathcal{R}_1(z;c) \;=\; E[\hat{\lambda}(z)'F_{t+1}] \tag{33}$$

where

$$\mathcal{R}_1^{infeas}(z) \;=\; \underbrace{\mathcal{R}_1(z;0)}_{zero\;\;complexity} \tag{34}$$

Theorem When $P \to \infty,\;\; P/T \to c$:

$$\mathcal{R}_1(z;c) \;=\; \mathcal{R}_1^{infeas}(Z_*(z)) \tag{35}$$

The Risk Of Doing ML

Theorem Suppose that $E[F]=0$. Then,

$$\lim_{P\to\infty,\ P/T\to c} E[R^F_{t+1}(z)] \ = \ 0. \tag{36}$$

Yet,

$$\lim_{P\to\infty,\ P/T\to c} E[(R^F_{t+1}(z))^2] \ = \ G(z;c) \ > \ 0\,, \tag{37}$$

where

$$G(z;c) \ = \ \lim_{T\to\infty, P/T\to c} \frac{1}{T} E[(F'_{t_1}(zI+B_T)^{-1}F_{t_2})^2] \tag{38}$$

for any $t_1 \neq t_2$ is given by

$$G(z;c) \ = \ (\xi(z;c)(1+\xi(z;c)) \ + \ z\xi'(z;c)+(\xi(z;c))^2)/(1+\xi(z;c))^2. \tag{39}$$

In particular, $G(z;c)$ is monotone decreasing in z and increasing in c.

Where Does The Risk Of Doing ML Come From?

To understand how the big data regime produces this intrinsic noise, consider a simple portfolio strategy that invests proportionally to the historical mean returns:

$$R^M_{t+1} \;=\; \bar{F}'_T\, F_{T+1}\,. \tag{40}$$

Then,

$$E[R^M_{t+1}] \;=\; E[\bar{F}'_T\, F_{T+1}] \;=\; E[\bar{F}_T]\, E[F_{T+1}] \;=\; 0\,, \tag{41}$$

under the assumption that $E[F] = 0$. Yet,

$$\begin{aligned} E[(R^M_{t+1})^2] \;&=\; E[(\bar{F}'_T\, F_{T+1})^2] \;=\; \operatorname{tr} E[\bar{F}_T \bar{F}'_T F_{T+1} F'_{T+1}] \\ =\; \operatorname{tr} E[\bar{F}_T \bar{F}'_T \Psi] \;&=\; \frac{1}{T^2} \sum_t \operatorname{tr} E[F_t F'_t \Psi] \;=\; \frac{1}{T} \operatorname{tr}(\Psi^2) \end{aligned} \tag{42}$$

If, for example, $\Psi = I$, this quantity equals $P/T \to c$. Thus, many minor estimation errors accumulate and generate non-trivial risk for the portfolio.

The Second Moment

Theorem

We have

$$E[(R^F_{T+1}(z))^2] \;\to\; \underbrace{\mathcal{R}_2^{infeas}(Z^*(z;c))}_{implicit\;\;regularization} \;+\; \underbrace{G(z;c)(1-2\mathcal{R}_1^{infeas}(Z^*(z;c))+\mathcal{R}_2^{infeas}(Z^*(z;c)))}_{estimation\;\;risk}, \tag{43}$$

where

$$\mathcal{R}_2^{infeas}(z) = \mathcal{R}_2(z;0) \;=\; \frac{d}{dz}\left(\frac{zA(z)}{1+A(z)}\right) \tag{44}$$

is the second moment of the return on the infeasible portfolio, $F'_{T+1}(\mathbf{z}I + E[FF'])^{-1}E[F]$, estimated using $T = \infty$.